# Meteorology for Actuaries – Part 2
## Climate and the El Niño Southern Oscillation

Gwendolyn Anderson, ACAS, MAAA

**The intertwining threats of climate change and catastrophe challenge society's ability to interpret shocks like recent hurricanes and wildfires. New capabilities have arisen in the recently expanded power of home computers which can now process vast databases; and in shared tools, such as R programming which offers calculation tools in combination with palatable "visual analysis" through plots and maps. Utilizing these technologies, this paper serves as a reference guide to weather analysis as it pertains to climate, and as regional climates relate to loss. A high level of detail in daily station records allows matching of specific weather measurements to losses in both time and location, lending ability to identify thresholds, durations, and combined forces leading to loss; further, changes in data or data quality can then be distinguished from shifts in climate. Physical explanations provide essential directions to begin exploration, focusing on an example of the phases of El Niño Southern Oscillation (ENSO) by which climate varies throughout the globe naturally, not only in extremes. The venture to discover climate's effect on losses becomes less daunting through pre-written modifiable code, sources for ENSO indices and other meaningful inputs, and a useful collection of tables and visual references.**

**Availability:** https://cran.r-project.org/
ftp://ftp.ncdc.noaa.gov/pub/data/ghcn/daily/

**Keywords**: Climate change, weather analysis, El Niño Southern Oscillation (ENSO), R programming, maps

**Abbreviations:**

| | |
|---|---|
| ENSO | El Niño Southern Oscillation |
| GHCN-D | Global Historical Climatology Network – Daily<br>ftp://ftp.ncdc.noaa.gov/pub/data/ghcn/daily/readme.txt |
| NOAA | National Oceanic & Atmospheric Administration |
| SST | Sea Surface Temperature |

**GHCN Weather Elements**

| | |
|---|---|
| `PRCP` | Precipitation |
| `SNOW` | Snowfall |
| `SNWD` | Snow depth |
| `TMAX` | Maximum temperature |
| `TMIN` | Minimum temperature |

`WIND*` elements are coded to include:

| | |
|---|---|
| `AWND` | Average daily wind speed |
| `WSF1` | Fastest 1-minute wind speed |
| `WSF2` | Fastest 2-minute wind speed |
| `WSF5` | Fastest 5-second wind speed |
| `WSFG` | Peak gust wind speed |
| `WSFI` | Highest instantaneous wind speed |
| `WSFM` | Fastest mile wind speed |

(* `WIND` is *not* an element abbreviation of GHCN-D.)

# 1. INTRODUCTION

Part One provided a basic backdrop of maps that can be instantly plotted in R language. Weather and one's own loss values may now be added to these backdrops. Part Two introduces daily meteorological data, publicly available through the Global Historical Climatology Network – Daily (GHCN-D). These vast datasets offer the level of detail suited to matching with weather-related losses in both time and location, easily accessed by R code with 8GB memory, a recent standard for most home computers. These combined advancements – memory, language, and data – expand the potential for exploring not only weather events but overall shifts of climate. This paper provides code, input sources and references, along with physical explanations of the weather phenomenon. Climate cycles are illustrated through an example of the El Niño Southern Oscillation. Many maps and plots in this paper are produced in R language from modifiable code provided in the appendix.

## 1.1 Research Context

As extreme weather events devastate North America, continually breaking records of a recent past, concerns widen over what seem to be pronounced changes in climate: is the potential for change understood well enough to simply prepare for the next storm? In absence of human industry, climate already changes naturally, with a myriad of interactions from diverse sources on multiple scales. Adding layers of complexity is the growing range of human activities that appear to impact climate systems, all while human skills and technologies advance in sync with nature's destruction. Portentous storms assert the need to utilize modern technologies to a timely advantage, to place state-of-the-art tools in reach to those with both common and uncommon skills.

### 1.1.1 Record Storm Losses

The costliest storms in United States history, those producing damages of $1 billion or more, are plotted below chronologically in actual unadjusted costs. A notable escalation of events occurred in the last three decades, disrupting the scale of catastrophic loss. Hurricane Hugo took a destructive inland course in 1989, followed in 1992 by Andrew which more than doubled record cost in three short years. Andrew led to insurer insolvencies, sending shock and a wake-up call through the industry. Professional leaders then turned to catastrophe modeling for answers, simulating the physical process of hurricane activity within trade secret models. This move proved effective in preparing financially for the spate of mega storms to follow.

By 2005, Katrina seemed to break the all the rules, striking levees and storm walls which had not been properly engineered to prevent vast flooding of the low-lying New Orleans area. Failed planning, it seemed, had quadrupled the former record set by Andrew.

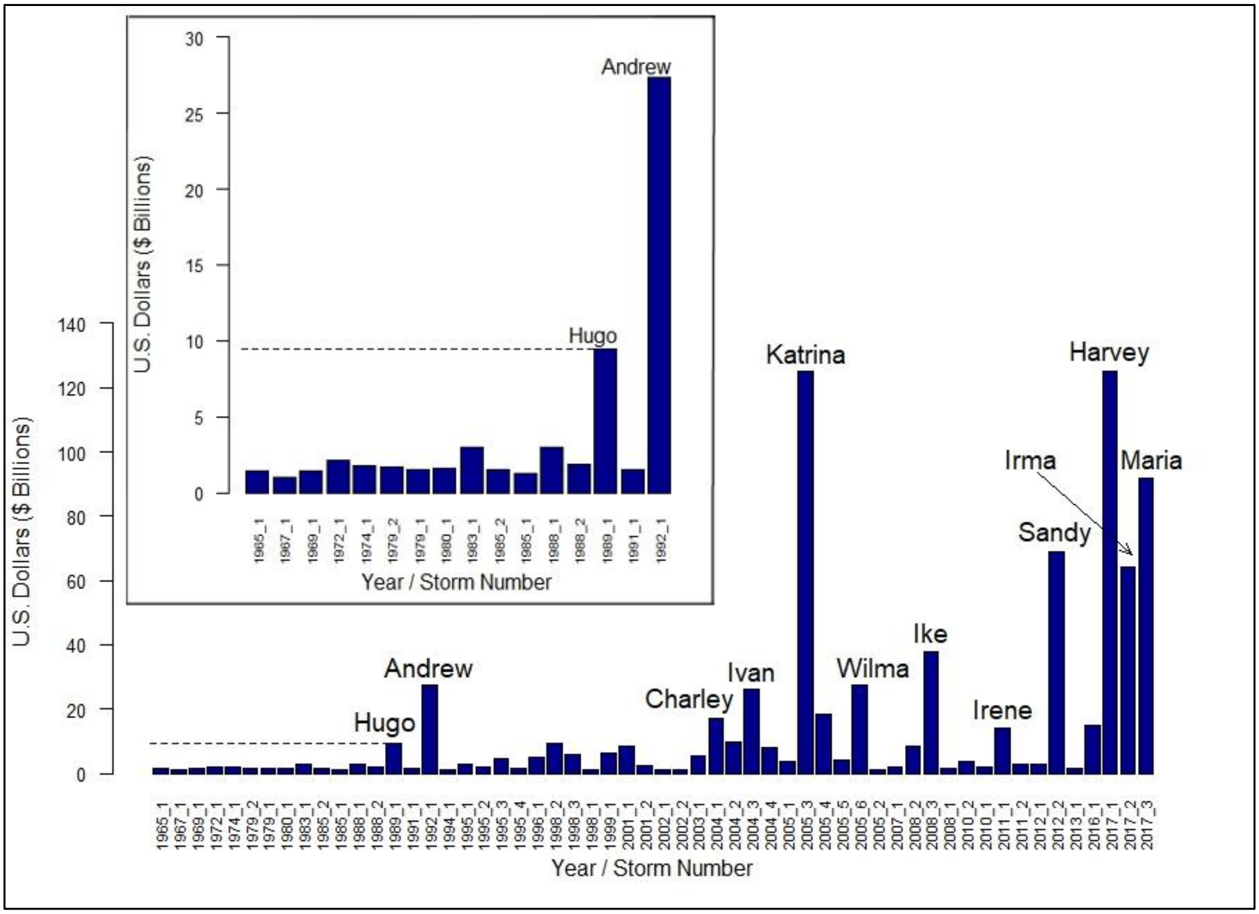



**Figure 1. Costliest Atlantic Hurricanes** - Katrina damages were a multiple of preceding "mega storms" due to failed engineering. In absence of adjustments, more recent storms seem to rescale catastrophes since the post-Andrew era. **Inset** - Andrew damages were unprecedented and took insurers by surprise. (All storms exceeding U.S. $1B in actual, unadjusted damages.)

Yet in 2012, Sandy struck the most densely populated area of the United States, unusually far to the north and late in season, again destroying property at multiple times the scope of Andrew. Five years later, the combined severity of three major land falling hurricanes in 2017 is unprecedented for a single season, with no poor engineering to blame. It is clear from these pictures that the costliest storms in all of history are also the most recent – imagine the shape had the Gulf Coast been protected against Katrina. These trends would appear to belie randomness and raise new questions surrounding severe weather and climate. In response to public concern, the role and sources of climate change might now be approached across broader disciplines.

### 1.1.2 A New Kind of Global Warming

Actuaries, modelers, economists and scientists alike are inclined to bring the historical loss record in line with present day conditions. Such adjustments include consideration for not only dollar inflation, but construction upgrades, migrations of population to coastlines, changes in levels of wealth near ocean fronts; and in the case of an insurance loss history, any changes in coverages or generosity of settlement. These financial insights alter the picture entirely: by various estimates of 'normalized' storm damages spanning over a century, the outcome appears a random process. Considering storms since 1900, the ICAT Damage Estimator [www.icatdamageestimator.com] ranks Katrina only as the fourth most damaging storm through 2012, and Sandy as eighth.

Hidden within the appearance of randomness is another process, recognized mainly in the scientific community, that might invite further refinements to views of weather-related losses. Somewhere in between the view of escalating catastrophes and the view of random losses, lies the natural force of *cyclical* climate change. The El Niño Southern Oscillation (ENSO) exerts major influence on the strength and timing of Atlantic hurricanes. Such cycles exist in absence of any human contribution to the atmosphere, and are irregular, reflecting a certain randomness of nature occurring in phases. These cycles may also be prone to change and may in themselves be subject to influences. How should climate be regarded if natural cycles might differ in frequency, duration or amplitude over future decades compared to the past century?

The same weather patterns or phases that influence severe weather events can be discerned more plainly in common weather elements like rainfall and temperature. These elements might attract less media attention than hurricanes, but will provide a far less volatile example of natural climate cycles. As a base illustration, Florida rainfall will be compared to ENSO indices in winter months, outside of the hurricane season.

The actuary, whose forté is prediction from limited data, might benefit from stepping into the shoes of the scientist, and might even tighten a few laces to fit a loss perspective. The weather data history is fraught with missing records and changes of stations whose measurements depend on elevation and surrounding conditions. If inflation, population, construction, wealth and coverage were not enough, changes in record keeping could also be mistaken for 'climate change.' The insurance industry may wish to weigh in on weather data collection now, to better account for climate shifts as they arrive.

A new trend in global warming may be in sight: a warming up to a cooperation in use of resources, from shared tools to shared understanding. Perhaps this trend may lead from strong varied opinions toward the exploration of facts and figures reinforced by the science that explains the physical phenomenon of weather.

## 1.2 Objective

Since research on weather and climate comes primarily from outside of the insurance sector, little focus is placed on loss estimates. Within the insurance industry, most research on these topics remains proprietary, limiting the public's grasp of the situation and limiting participation by those who might strengthen the climate conversation. This paper seeks to remove limitations to analysis. At most, the boundlessness of relationships to be explored might be recognized. At least, a highly detailed resource of element measurements should illuminate the sparseness of record available by which to identify climate shifts.

This paper provides tools and references to accompany the vast daily station data of the Global Historical Climatology Network (GHCN), along with an understanding of the physical processes of weather as insight to the analysis of weather peril losses. A framework is provided through code and useful references, with an example of the El Niño Southern Oscillation and Florida winter rainfall. Broad paths may be explored through this data set, whether the direction one wishes to pursue is global or focused within a unique region.

## 1.3 Outline

Background and Methods, Section 2, suggests methods for matching daily weather data sets to losses, through focus on damageability thresholds, durations, and interactions of weather elements that lead to loss. Beyond a programming method, a background in the physical phenomenon of weather guides interpretation of the data set and provides a basis for analysis. The description begins with the source of weather: the heat of the equator. Next, the motion of weather enters through the atmospheric circulation by which the heat is redistributed on earth. This leads to the core phenomenon to be covered, the El Niño Southern Oscillation (ENSO), with its pronounced influence on weather patterns in parts the globe distant from its origination around the equatorial Pacific. Sources are provided for indices that measure different oceanic regions of the ENSO phase by Sea Surface Temperature (SST), pressure, and other attributes. The time scale of 'climate' is differentiated from that of 'weather,' and cycles are recognized as a climate determinant. The short history of meteorological records gives insight into the sparseness of measurement available to compare climate over time, in spite of large data sets available today. The actuary's unique capabilities where data is lacking could be constructive contributors in the climate arena.

Results, Section 3, presents example findings from code output, including United States maps of station locations; and choropleths, anomalies color-coded by state. Some summaries of missing records and data changes are given by year. Florida winter rainfall is shown to correlate well to some ENSO indices and not to others.

Code to process the GHCN-D data is provided in the appendix. The code is intended for modification to the level desired.

## 2. BACKGROUND AND METHODS

The relationship between weather perils and insurance losses may be explored by linking a history of loss and exposures to the meteorological data. An approach is desired that would isolate the types of weather events leading to loss. Some background in the physical process of weather provide necessary insights to the analysis, bringing awareness of the natural climate cycles of the El Niño Southern Oscillation. Data quality and completeness require attention so that data changes not be mistaken for 'climate change.'

### 2.1 Thresholds

A high level of detail in daily station records allows matching of specific weather measurements to losses in both time and location. This detail lends the ability to establish thresholds at which losses are likely to occur. Thresholds tend to represent physical phenomena, such as zero degrees Celsius at which water freezes, or wind speeds that topple trees.

Durations of extreme weather are also relevant, and can be tracked daily up to the time of loss, such as low levels of precipitation eventually leading to crop loss. Combined forces may lead to damages, such as drought accompanied by high temperatures. Damaging interaction of weather elements may be intertemporal, such as drought-inflicted regions becoming susceptible to fires or mudslides with higher temperatures or rainfall, respectively. Thresholds should be expected to vary by region, for instance, Seattle with its immense drainage capacity may withstand multiples the rainfall of flood-prone Charleston.

Thresholds and durations cannot be extracted from monthly summaries, and loss events cannot be pinpointed in data sets that have been gridded in rectangular areas encompassing multiple stations. A maximum monthly temperature or average monthly temperature is not useful. Summaries that count threshold values can be created from daily data while retaining the source detail. Care must be taken to adjust for various changes to daily record keeping over time.

For the purpose of measuring climate change, standard deviation anomalies from a selected base period average serve as straightforward and meaningful measures. The anomaly will usually be calculated for a summary period, such as a month or year, compared to some longer base of 30 Januaries or 30 full years, for instance. These figures give an intuitive sense of fluctuation across time with appropriate scaling for the selected region; a large anomaly of rainfall in the desert will represent a small quantity in comparison to the same anomaly in the tropics. In absence of climate change, anomalies will take values spread about zero according to the stable underlying distribution of the element. Relative values like temperatures will be normally distributed while quantities like precipitation, bounded below by zero, will be skew.

Some regions suffer no loss from large deviation weather events while others regions hover at the edge of the climate extremes where disasters occur. Anomalous weather events could impact loss if present climate extremes of a region are close to loss thresholds. Attention should be given to cycles and shifts of climate in regions where loss thresholds have been crossed or where near-threshold weather patterns can be identified. The distribution of the weather element could be tracked over time or compared against the base period.

One familiar threshold guide is the Saffir-Simpson scale, which assigns a level of damage to hurricane categories by wind speed. The types of damages will vary by region and by the types of buildings in the region. The same damageability scales would clearly not apply in a country with building standards inferior to those of the United States.

**Table 1. Saffir-Simpson Hurricane Wind Scale**

| Category | Sustained Winds | Damages |
|---|---|---|
| 1 | 74-95 mph | Very dangerous winds will produce some damage |
| 2 | 96-110 mph | Extremely dangerous winds will cause extensive damage |
| 3 | 111-129 mph | Devastating damage will occur |
| 4 | 130-156 mph | Catastrophic damage will occur |
| 5 | 157+ mph | Catastrophic damage will occur with increased severity |

Station detail is especially critical for ascertaining data completeness and quality, a realization erased by most summaries and grids. A common practice before 1982 was to assume missing daily quantity records were zero, a critical value for tracking drought. Thresholds cannot be reliably identified on days where values are left blank or assumed zero, unless, of course, a method is employed to generate values providing better information than the entered records.

With some R code already written and ready to run, delving into the data should be straightforward. The analysis is perhaps only as complicated as the weather.

## 2.2 The Source of Weather

A grasp of the concept of climate and its potential for change stems from understanding the physical source of weather: heat. The basics of daily and seasonal weather, which derive from heat and movement, explain the mechanisms of the El Niño Southern Oscillation, or "ENSO," with its varying phases of impact on regional climates.

The intense heat from the sun's rays near the equator seeks to equalize itself across the earth through winds and currents, all while the earth is engaged in two circular motions, rotating on a tilted axis while simultaneously orbiting the sun. People speak often of the "sunrise" and "sunset," and of changes in weather they feel which may be swift and drastic. Yet holding constant is the *imperceptive quality* of the underlying phenomena of motion around both an axis and a "stationary" sun, a sun that neither rises nor sets. While of little consequence to weather, the entire solar system including the sun and earth are actually moving through space around the Milky Way Galaxy in a third grander orbit. So the earth is in orbit, along with other planets, in a spiraling motion through space about a moving sun. The solar system's orbit might only impact the earth's climate over tens of millions of years. What is more, the Milky Way Galaxy is itself in orbit with other galaxies.

While essentially "sitting still" at her desk, an analyst could make fairly precise calculations of the earth's rotational motion based on latitude, all while feeling nothing of the earth beneath her speeding around and around at a staggering rate of over 800 miles per hour. This figure would increase to over 1,000 mph were she located near the equator. Simultaneously, she is orbiting the sun at 66,700 miles per hour so that in one full turn of the seasons, the distance traveled amounts to 584 million miles. The earth makes one complete revolution, completing a "sidereal day," in about four minutes short of 24 hours. Over four thousand miles of *orbit* completes each cycle of the "solar" day in an average of around four minutes – astounding speed! Since the earth's orbit it elliptical, the time and distance to complete a solar day varies with closeness to the sun. By the earth's dramatic motion in space, the state of heat inequality is driven by a rapid change of position.

With one half of the spheroid planet always illuminated, the surface of the earth travels *thousands* of miles in a single day's rotation to distribute heat evenly around it like a chicken roasting on a spit, which translates into a seemingly trivial differentiation of temperatures: cooler in the morning and at night compared to afternoon. The lag of several hours in respectively the warmest and coolest temperatures of the day following midday and midnight, comes from the magnificent ability of the earth's surface and atmosphere to store and slowly release heat energy. The *hundreds of millions* of miles in revolution through the solar system differentiates seasons – but only due to the slight tilt of the earth on its rotational axis. In its elliptical orbit, the earth's varying distance from the sun does not significantly influence temperatures. Rather the angle of the sun's rays decides intensity. A common illustration is a flashlight directed straight at a wall: moving the distance of the beam's source forward and away scarcely influences the light's intensity compared to angling its direction to a slant – the angle diffuses the brightness. Were the earth to spin straight up and down on a vertical axis while orbiting the sun, even hundreds of millions of miles could not produce a January distinguishable from June.

The atmospheric circulation on earth – a large scale movement of air distributing thermal energy across the earth's surface – can be described by the process of "convection." Convection is a circular motion of molecules within fluid, where fluids encompass both liquids and gases such as air. With a difference in temperature, hotter material rises while the colder sinks with gravity. In a room, hot air rises to the ceiling. On earth, the convective process occurs within the troposphere both latitudinally, from the equator to the poles, and longitudinally across the equator. From the equator to the poles, the decrease of solar intensity with latitude sets convective circulation patterns into motion. Along the equator, a difference in temperature arises between land and ocean because of the substantial difference in the amount of heat these surfaces types absorb and emit.

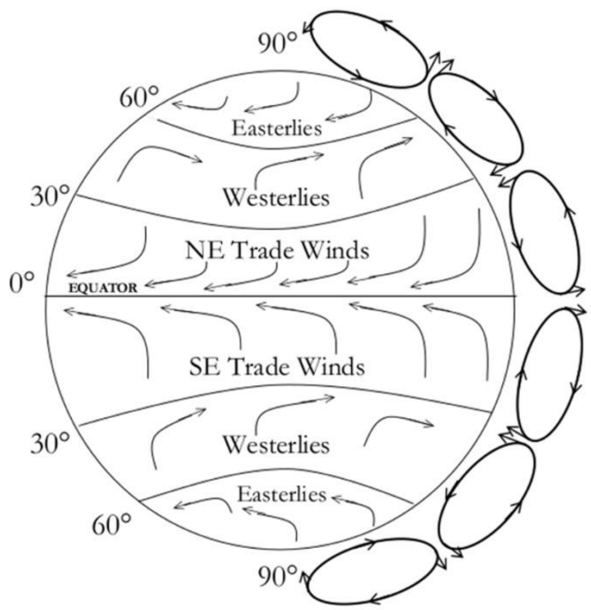



**Figure 2.** This illustration shows "idealized" patterns of ocean currents and the six convective cells which wrap around the globe within the troposphere, the lowest level of the atmosphere, where weather occurs. The rotation of the earth produces ocean currents flowing in opposite directions and breaks in the convective circulation loops at approximately 30° and 60° north and south.

Were the earth to stand still on its axis, cold winds would blow from the poles to the equator across its surface while hot air would rise at the equator in a convective circulation towards the poles. Rotation enters this equation with an elaborate influence, generating six segments of "idealized" wind directions that deviate from the theory, as all weather does, with changes in terrain and a profusion of random disturbances and interactions. Nearest the equator are the easterly (i.e. "from the east") trade winds which early merchant ships sailed, ranging from about 0° to 30° north and south. In both hemispheres from around 30° to 60° are the westerlies (i.e. "from the west") by which those ships made their return voyages, and at roughly 60° to 90° the circulation again reverses to easterly polar winds. Were it not for the complex circulation patterns arising from the earth's rotation, intercontinental trade could not have taken place by sail and oar. The force of the earth's rotation is strongest at the poles and weakens towards the equator, where seafarers could become trapped in the calm of the "doldrums." The circulation pattern along the equator, where rotation produces no force of deflection, is known as the "Walker circulation." At the equator, easterly winds across the wide open Pacific, in concert with the Walker circulation, give rise to the El Niño Southern Oscillation. Around the 30° latitude lines, subsiding dry air of the convective cells generates the desert regions in bands across Africa and Australia. From as far away as the farthest eastern end of the African deserts, dry subsiding air stirs winds that may continue to travel from east to west across the hot African land deriving strength to propel still further west across a warm Atlantic and morph into some of the most powerful hurricanes striking the eastern United States. This storm pathway illustrates the weather system is truly massive.

## 2.3 The Atmosphere

The atmosphere would be "paper thin" if the earth were scaled on the size of a basketball. The phenomenon of weather occurs only within its very base layer, the troposphere. Mount Everest, at just over 29,000 feet elevation (about five and a half miles), sits in the upper troposphere. The final layer of atmosphere ends about 6,200 miles from the surface which would only be a twelve hour flight, could an airplane traverse the thinning air.

Cold temperatures compress molecules, so that colder air is denser with less movement of molecules. Areas of high pressure – which essentially originate from coldness – move towards areas of low pressure – similarly defined by warmth – so the pressure differences from unequal heating near the earth's surface give rise to winds. The height of the troposphere varies with temperature and changes with seasons: at the equator it may extend as high as twelve miles while the winter poles may compress the layer to seven miles.

The earth's atmosphere is naturally comprised of gases. In dry air, without consideration of water vapor, the composition is roughly 78% nitrogen and 21% oxygen, gases which allow heat leaving the earth's surface to pass through and escape into space. The remaining roughly one percent of the mixture includes a very small proportion of "greenhouse gases" – gases typically measured in parts per million or billion which absorb heat released from the earth and trap them near the surface. These gases include carbon dioxide, methane and nitrous oxide. Water vapor is another greenhouse gas present in varying proportions by region, making up nearly 4% of the troposphere's gases in tropical regions near the equator, but closer to 1% near the poles. The proportion also varies through the natural cycles of cloud formation and precipitation. Without naturally occurring greenhouse gases, scientists estimate the average temperature at the earth's surface would drop from 59°F to 0°F. The mixture is precise: with less than 16% oxygen content, ordinary fires would not burn; while high oxygen concentrations would aggravate combustibility. Therefore these molecular elements are precious to life on earth, and no more detectable to us than the motion of the earth beneath our feet. Yet imagine in its entirety, only a few miles outside the range of sight and rotating along with us, this thin invisible atmosphere is enough to disguise the hurling high speeds of the earth's rotation and orbit! This illustrates that the climate system, while massive, is also meticulously detailed.

Scientists agree that adding greenhouse gases to the atmosphere will raise surface temperatures. The warming effect of recent history is best illustrated in the award-winning documentary "Chasing Ice" in which photographer James Balof chronicles the rapid melting of glaciers. Charles Keeling began recording carbon dioxide levels in the atmosphere at Mauna Loa Observatory beginning in 1958, noting seasonal variations of concentrations in the atmosphere; by 1961 he issued the first warnings of anthropogenic contributions to the greenhouse effect. Roger Pielke Sr. stirred controversy in 2007 by claiming carbon dioxide accounts for only 28% of human-caused warming, stressing the remaining 72% is still human caused.

Large bodies of water absorb and release heat at a much slower rate than the atmosphere or ground terrain, requiring over a thousand times the energy to heat as the same volume of air. The upper ocean near the surface can store approximately 30 times the heat as the atmosphere immediately above it. Interaction between water bodies and the atmosphere also creates sea breezes. These phenomena lead climates near coasts and large lakes to be more temperate than areas inland. The ocean is a gigantic sink for atmospheric warming, the effects of which may not be felt so well on land until the ocean has reached its full capacity for absorption.

Other human activities and natural forces can cause temperatures to rise, or fall, and climate change collectively refers to all types of changes to regional climates or long-term weather patterns and extremes, not only heating, but cooling or changes in precipitation or winds. For instance, deforestation releases carbon to the atmosphere but further alters surface reflectivity from greener to drier while removing the valuable balancing process of photosynthesis by which carbon dioxide is converted with sunlight into oxygen. Forests can suddenly be replaced by agriculture or housing tracts; water use, land use, and controlled burning can all immediately influence climate. City streets of asphalt have induced the "urban heat island effect," an effect that can be counteracted with the numerous benefits of roof gardens. Nuclear power plants raise the water temperature of adjacent lakes that supply water to cooling towers. Natural volcanic eruptions spew carbon and particulate matter into the atmosphere, typically cooling the earth for several years from the high reflectivity of particles. Particulate matter from all types of pollution, even dust rising from cleared fields, assists storm clouds to grow larger and form into more powerful storms. While greenhouse effects are described as slow and gradual, many types of climate change are more immediate including the natural cycles of ENSO.

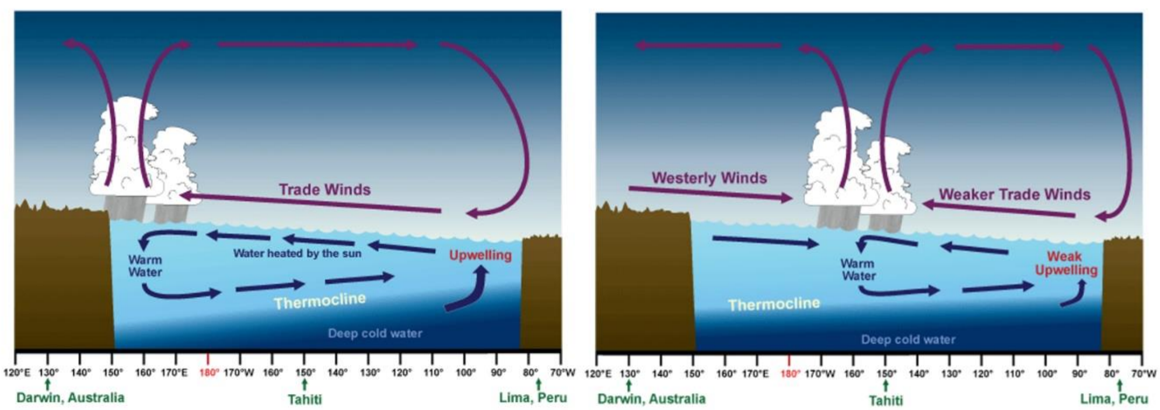
## 2.4 El Niño Southern Oscillation

The Pacific is the largest body of water in the world, twice as large as the Atlantic and far deeper. Its expanse across the hot equatorial region wraps nearly half the earth's circumference, spreading the canvas for the brush strokes of the El Niño Southern Oscillation, or simply ENSO. Temperature and pressure will typically differ substantially from one end of the Pacific to the other. The tropics of the western Pacific hold some of the hottest water in the world's oceans: surface temperatures may warm to around 84°F covering an area the size of Australia. At the Peruvian coast, temperatures may be as cold as 60°, uncharacteristically low for the tropics. Yet the sun's rays are of equal strength all across this equatorial region.

Motions and attributes of oceans are not separated from atmosphere; rather the two interact with "positive feedback loops" by which changes are amplified, pushing away from equilibrium to invite instability. The atmosphere responds to disruptions quickly in time scales of days to weeks, while the ocean reacts more slowly, over months to years. The El Niño Southern Oscillation is a single large-scale coupled interaction of atmospheric pressure and ocean temperature across the Pacific Ocean, stretching from the coast of South America at Ecuador and Peru in the east to Indonesia and Australia in the west. "Southern Oscillation" refers to the "seesaw" effect in atmospheric pressure between the eastern and western Pacific: when pressure at one end shifts to lower than normal the other end will become higher than normal. "El Niño" refers to ocean warming across the Pacific equator which occurs together with the dominating shifts of pressure. These shifts in the tropics can exert powerful influence on global weather.

Beneath an evenly intense sun, a striking contrast in surface temperatures arises at opposite ends of the central Pacific. Five major contributors emanate primarily from the earth's rotation: (1) heating by both sun and warm air as water is pushed westward along the equator by trade winds, (2) upwelling along the equator by the same motion of the trade winds, (3) cold upwelling at the coast of Peru, (4) warm downwelling at Indonesia, and (5) the change in the depth at which colder waters lie from the surface, across the equator. The underlying mechanisms deserve elaboration before considering how a reversal takes place.

As winds blow across the surface of any body of water, the turning motion of the rotating earth will cause the water to spiral so that it moves overall perpendicularly to the wind direction. In the southern hemisphere, water is deflected to the left of the wind direction; and in the northern to the right. Winds blowing towards the equator from both the north and the south turn towards the west. As water is displaced along the equator from either side, water from below the surface rushes in to replenish the space. Winds blowing northward along the coast of Peru similarly produce an upwelling, where temperatures near the ocean's surface are cold. The opposite occurs at Indonesia and the other land barriers of the Maritime Continent, where westward winds produce a downwelling of warm surface waters according to the direction of the earth's rotation. Sinking warm waters push the colder basin waters below down to even further depths. Note that the ocean is stratified: water near the surface is warm from various influences such as the sun's heat, evaporation, and mixing winds; while deeper waters are still and cold. The "thermocline" lies in between, a thin dividing layer in which temperatures drop quickly through a shallow depth. The sinking of warm waters at the western Pacific encourage a downward slope to the thermocline from east to west. The waters upwelled along the equator by "the trades" increase in warmth moving west as the cold lower layer slopes down further and further below surface.




Source: NOAA Jetstream

**Figure 3. El Niño Southern Oscillation.** The 'normal' state of the Pacific Ocean is illustrated on the left; but when conditions are amplified the same pattern become a La Niña event. An El Niño event is illustrated on the right.

Warm surface water pushed westward by the trades eventually encounters barriers in the land masses of Australia and Indonesia, where it literally piles up. Over time the western sea level may gain 20 inches elevation, forming a mound of water visible from space. This view serves as an assessment of the ENSO phase. The slope of the ocean's surface, then, opposes the slope of the thermocline. The heated water that reaches the Maritime Continent of Indonesia evaporates from the ocean, condenses into rain clouds, and pours out tropical rain storms, fueling upper level winds. Every year, over 100 quadrillion ($10^{17}$) gallons of water evaporates from the ocean, mostly around the tropical equator, with about 90% of the precipitation falling over the ocean. Rising warm air travels through the troposphere eastward back across the equator and then settles in a convective loop, reinforcing the westward trade winds along the surface.

Awareness of a reversal in the usual pattern originated in Peru. Ordinarily, winds blow northward along the coast of Peru stirring up cold waters, replacing depleted surface waters by rich nutrients from deep basin waters – that feed vibrant fish populations – which in turn sustain bird populations – whose droppings provide fertilizer to the agricultural sector. A seasonal transformation of an inconsistently warm current entering this coastal region was first identified by Peruvians at Christmastime as El Niño, *the boy* or *the Christ child*. A strong El Niño event can devastate Peruvian fisheries, impair agriculture, and induce rain storms that flood the coastal regions.

When trade winds are brisk, coastal upwelling is strong along Peru, and the thermocline is steep, an amplified phase of colder eastern sea surface temperatures may be referred to as La Niña, or *the girl*. The same conditions at a lesser strength are considered "neutral," or the "normal" state of the Pacific – sometimes called La Nada, *the nothing* – a state which does not prompt severe weather.

An El Niño event always begins with pressure changes, namely, a lessening of the pressure gradient between the eastern and western Pacific. Since winds blow from high to low pressure, this leveling of pressure weakens the trade winds that have driven water to pile up towards the west. The heated water will then slosh back in a countercurrent that sends the excesses of warm water across the Pacific. The central and eastern regions of the Pacific waters warm near equal to the western temperatures, repositions the intense rainstorms away from Indonesia towards the central or eastern Pacific, and shifting large scale wind patterns in turn. Pronounced phases of ENSO – El Niño and La Niña alike – are known for diverse consequences of extreme weather at near and distant regions of the globe, *sometimes* with opposite impacts to one another. All of the effects together do not amount to true opposites considering some arise from a shift in the region of predominant precipitation, a location change which is not an opposite.

The term "El Niño" has come to signify an amplified cycle which typically occurs on intervals of three to five years, historically from two to seven years. Variation is not only in frequency and strength but also duration which may span several months to a few years. La Niña is especially well known for enhancing Atlantic basin hurricane activity. Within the troposphere where weather occurs, various wind speeds and directions may occur for several miles above the ground known as "vertical wind shear" which when strong, can topple hurricanes or stifle their formation. La Niña conditions foster an evenness along altitudes favorable to hurricane formation and survival, that is, a weakening of vertical wind shear. Other consequences of El Niño and La Niña are shown in the maps following.

What triggers the pressure gradient to lessen, unleashing an El Niño event, remains a scientific mystery. There may not be one precise answer since weather is influenced by numerous factors characterized by random occurrences, and further by interactions and also feedbacks. Ambient air pressure is constantly changing, and even while the pressure changes are measurable, the sources of change may not be discernible. Random distortions to any number of usual weather patterns or combinations thereof could eventually lead to shifts of pressure at the equator: sudden bursts of opposing winds, sub-surface waves, changes in salinity from the sinking of salty waters along the equator, or distant elements such as mountain snowpack or glacier ice, could shape valid hypotheses. This mystery beneath recurring large-scale global weather patterns illustrates that the climate system, both massive and detailed, remains largely "over our heads."

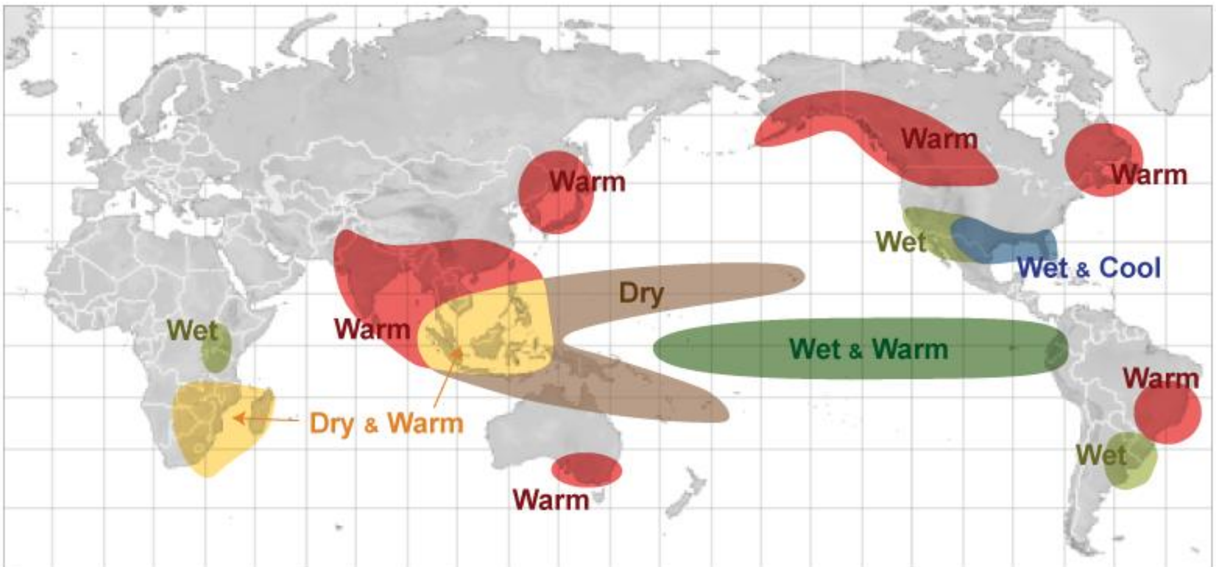
**Table 2. Summary of ENSO event characteristics** – the phases of the El Niño Southern Oscillation may be summarized by a few characteristics. When La Niña conditions are present but are mild, not amplified, the phase is neutral and global weather patterns are not influenced.

| La Niña | El Niño |
|---|---|
| Strong upwelling of cold deep basin waters at coast of Peru | Weaker upwelling along Peruvian coast, and upwelling of warmer waters |
| Steep thermocline with cold water nearer to surface | Less slanted thermocline |
| Strong easterly trade winds | Weakening easterly trade winds |
| Warm western Pacific and cooler eastern and central Pacific | Central to eastern Pacific assume warmer temperature, nearer that of western Pacific |
| Region of persistent precipitation is over warmest water near Indonesia | Region of persistent precipitation is shifted, over warmest water near central Pacific |
| High sea level pressure in eastern Pacific differs from low pressure in western Pacific – strong Walker circulation | Sea level pressure in eastern Pacific lowers near to level of western Pacific – weakening Walker circulation |

**Figure 4. Regional Weather Impacts of El Niño Southern Oscillation** – El Niño and La Niña – winter and summer seasons. These four maps provided by the NOAA serve as excellent reference to the regional effects of natural climate cycles in weather data specific to the ENSO phenomenon, and may be consulted for planning climate phase analyses by location and time of year.

**(A) El Niño - winter season**

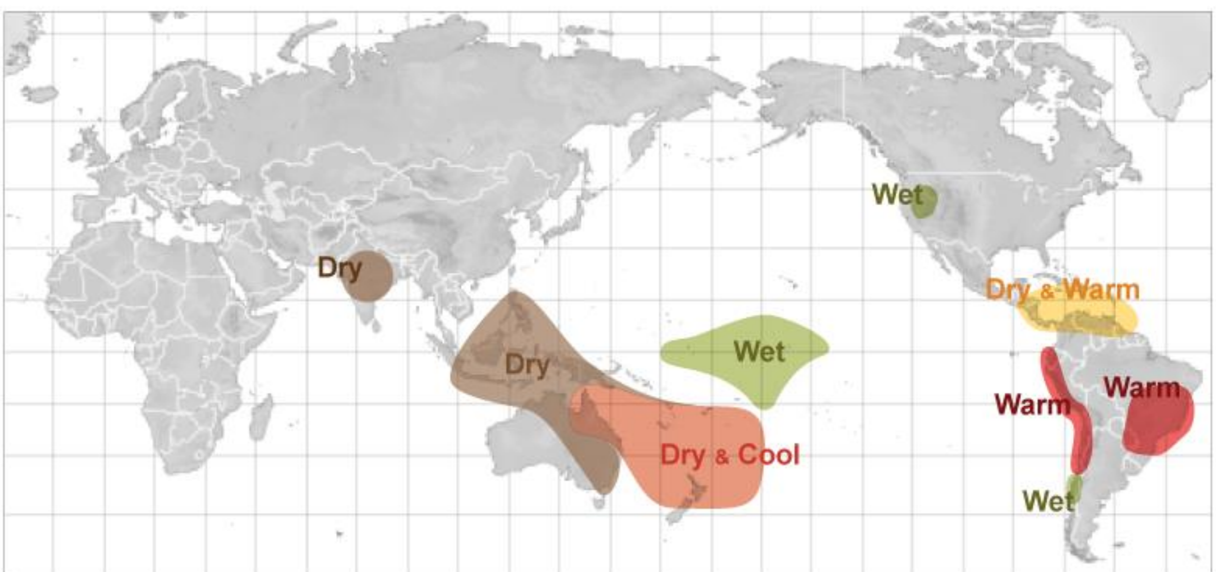
*El Niño effects during December through February*




Source: NOAA Jetstream

**(B) El Niño - summer season**

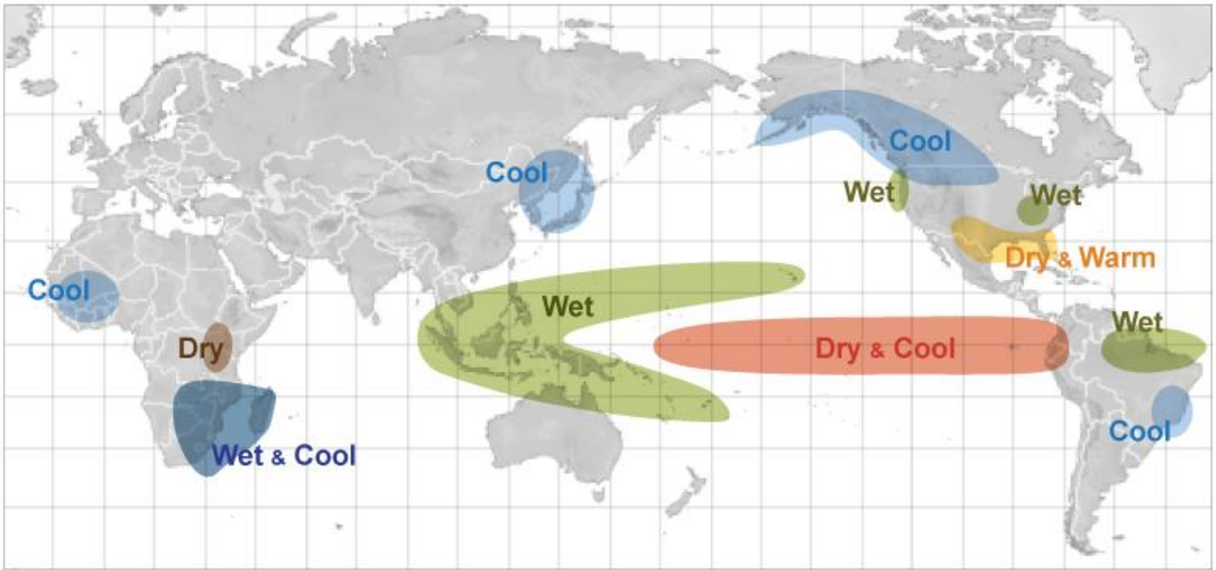
*El Niño effects during June through August*




Source: NOAA Jetstream

**(C) La Niña - winter season**

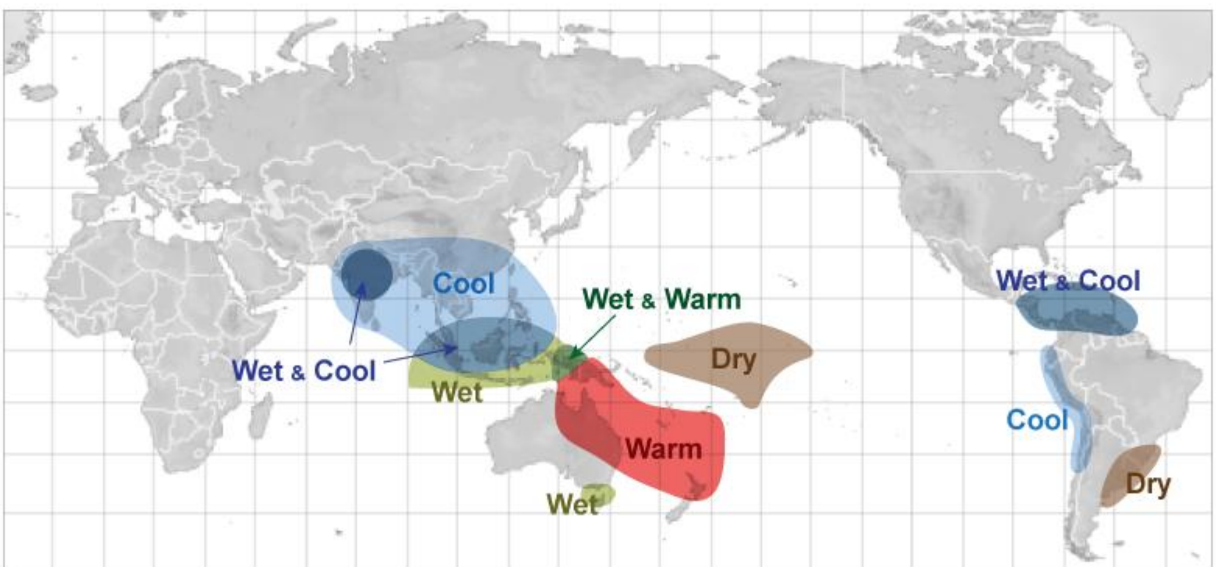
*La Niña effects during December through February*




Source: NOAA Jetstream

**(D) La Niña - summer season**

*La Niña effects during June through August*




Source: NOAA Jetstream

## 2.5 ENSO Indices

The original indices tracking the phase of ENSO are named by the ship tracks that originally recorded sea surface temperatures (SST) across this equatorial region of the Pacific, beginning with Niño 1 and 2 near the coast of Peru where destructive forces of the ENSO phenomenon were first witnessed. Niño 3 extends across an eastern equatorial band of the Pacific, reflecting the later realization of a farther reaching phenomenon. Niño 4 covers the tropics to the west, and Niño 3.4 is measured in a midregion overlapping Niño 3 and 4. The Niño indices are recorded most commonly as average monthly SST and are also given weekly, and further as anomalies from a base mean SST value. The more extreme colder temperatures relate to La Niña events, the warmer to El Niño.

The Ocean Niño Index (ONI) is derived from the Niño 3.4 SST as rolling three month periods (Jan-Feb-Mar, Feb-Mar-Apr, etc.). The Trans-Niño Index (TNI) is derived in a different manner combining Niño 1 and 2 with Niño 4. The TNI considers that the difference in SST on opposite sides of the Pacific better reflects the phase for certain purposes, and takes the standardized Niño 1 and 2 minus the Niño 4 with an additional standardizing adjustment; specifically, a five month running mean is applied and then standardized using the 1950-1979 period. The regions of measurements for ENSO indices are shown in the map below.

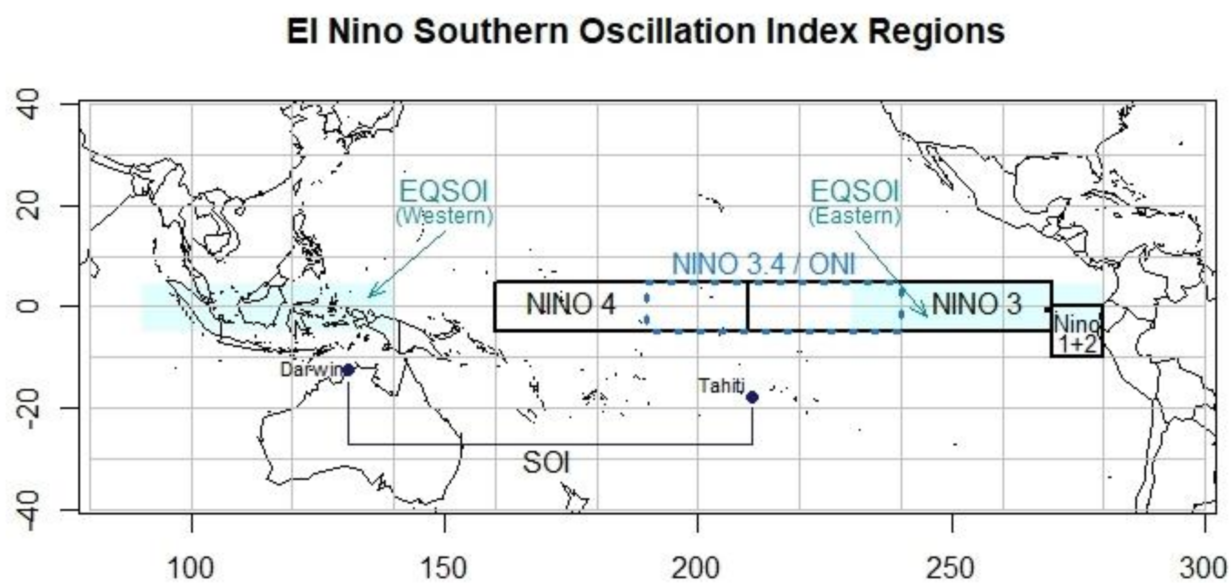



**Figure 5. ENSO Regions** – regions where the phase of ENSO is measured by SST or pressure are shown in a Pacific-centric map. The TNI is based on Niño 1+2 and Niño 4 while 'BEST' is based on Niño 3.4 and the SOI.

**Table 3. ENSO Index Coordinates** – the coordinates where ENSO indices are measured are given in Atlantic-centric coordinates (-180° to 180°) and Pacific-centric coordinates (0° to 360°)

| ENSO Index | Atlantic Coordinates | Pacific Coordinates |
|---|---|---|
| Niño 1+2 / TNI (east) | 0°-10°S, 90°W-80°W | 0°-10°S, 270°E-280°E |
| Niño 3 | 5°N-5°S, 150°W-90°W | 5°N-5°S, 210°E-270°E |
| Niño 3.4 / ONI / 'BEST'(i) | 5°N-5°S, 170°W-120°W | 5°N-5°S, 190°E-240°E |
| Niño 4 / TNI (west) | 5°N-5°S, 160°E-150°W | 5°N-5°S, 160°E-210°E |
| EQSOI (west) | 5°N-5°S, 220°W-270°W | 5°N-5°S, 90°E-140°E |
| EQSOI (east) | 5°N-5°S, 80°W - 130°W | 5°N-5°S, 230°E-280°E |
| SOI / 'BEST'(ii): | | |
| Darwin, Australia | | 12.4634°S, 130.8456°E |
| Tahiti | | 17.6509°S, 210.5740°E |

Further indices exist to track the ENSO phase without SST measures. The Southern Oscillation Index (SOI) records the large-scale fluctuations in pressure between the western and eastern Pacific, at the locations of Darwin, Australia versus Tahiti. The pressure differential is associated with heat in the atmosphere as opposed to the surface water of the ocean, and the atmospheric pressure gradient is prone to change much more swiftly than ocean temperatures. The SOI is more negative during an El Niño event, where pressure in the eastern Pacific lowers nearer to that of the western Pacific. The Equatorial SOI is another measure based on pressure, but instead of relying on two distinct points observes averages across larger regions, over Indonesia and off the coast of Ecuador.

The Multivariate ENSO index (MEI) combines several characteristics into one index. Its calculation considers the six main observed variables over the tropical Pacific: sea-level pressure (P), zonal (U) and meridional (V) components of the surface wind, sea surface temperature (S), surface air temperature (A), and total cloudiness fraction of the sky (C); calculated in rolling bimonthly periods (Jan-Feb, Feb-Mar, etc.). Various index measures track different characteristics of the ENSO phase, so they will serve as unequal indicators to climate effects in various regions of the globe. Klaus Wolter of the NOAA describes the relevance of the MEI, in relation to other indices, as follows:

> "Why do I believe that the MEI is better for monitoring ENSO than the SOI or various SST indices? In brief, the MEI integrates more information than other indices, it reflects the nature of the coupled ocean-atmosphere system better than either component, and it is less vulnerable to occasional data glitches in the monthly update cycles. Now, if you are interested in ENSO impacts in a very specific part of the world, I would suggest that you obtain other ENSO indices as well and establish which one best fits your needs. For instance, in Australia, Darwin sea level pressure and/or the SOI may be more appropriate than the MEI. My claim here is that the MEI does a better job than other indices for the overall monitoring of the ENSO phenomenon, including, for instance, world-wide correlations with surface temperatures and rainfall."

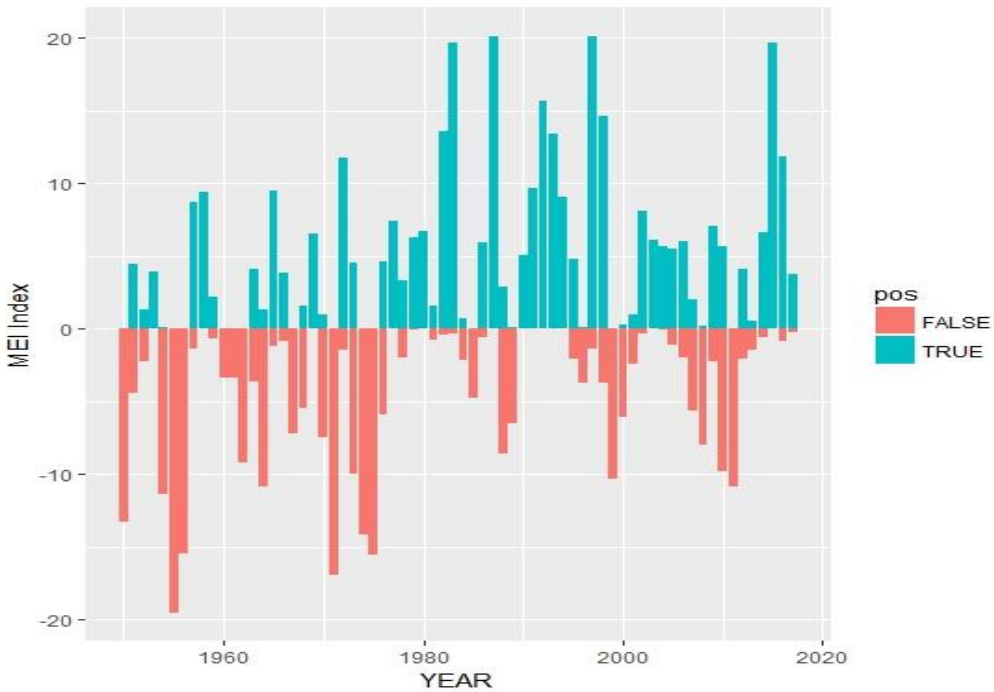



**Figure 6. Phases of the MEI.** Multivariate ENSO Index since 1950.

Indices tracking ENSO phases are available online at these NOAA sites:

| | Website Address | Indices (format: Wide or Long) | from |
|---|---|---|---|
| (I) | www.cpc.ncep.noaa.gov/data/indices/ | Niño, ONI (L); SOI, EQSOI (W) | 1950's |
| | | Niño Weekly (L) | 1990's |
| (II) | www.esrl.noaa.gov/psd/gcos_wgsp/Timeseries/ | Niño (W); SOI (W) | 1870's |
| (III) | www.esrl.noaa.gov/psd/data/climateindices/list/ | Niño, ONI SOI, TNI, BEST, MEI (W) | 1950's |
| (IV) | www.esrl.noaa.gov/psd/enso/mei/table.html | MEI (W) | 1950's |

(I) NOAA National Weather Service – Climate Prediction Center – Monthly Atmospheric & SST Indices
(II) Global Climate Observing System – Working Group on Surface Pressure

NOAA– Earth System Research Library – Physical Sciences Division – Climate Indices – Monthly Atmospheric and Ocean Time Series
(III) NOAA– Earth System Research Library – Physical Sciences Division – Multivariate ENSO Index

## 2.6 Climate versus Weather

Weather typically describes short-term phenomena while climate describes the long-term weather conditions that predominate a specific region. A "climatological normal" is an average of a weather element over 30 years, which serves as a base for comparison. For scientific purposes, climate is usually defined by a 30-year period; for some purposes, the base climate period chosen might span 40 to 100 years. The definition of climate includes not only the long-term averages and typical variations in the elements, but further places emphasis on the extremes experienced over the full range of the selected base period. A "very hot day," then, describes weather, while "the hottest day in London since 1976" designates a boundary for one major city's climate.

The common 30-year scope implies that weather is expected to fluctuate to a certain extent, from one year to the next, and variations in this range would not constitute climate change. The assumption that three decades would cover the irregular fluctuations of the El Niño Southern Oscillation might also be implied, since this is the major cyclical climate factor for some regions. But because of the myriad of interactions among climate variables, not only is ENSO a source of natural climate variations, ENSO is itself susceptible to change. A base climate period might be more closely examined for trends, cycles, and shocks. Irregularities might be taken to another level of comparison and adjustment when considering future loss potential.

For the examples of this paper, the years 1961-1990 are selected as a base period for climate. This period corresponds to the earliest 30-year term at which instruments are considered reliable and consistently gauged. Care should certainly be taken in relying upon analyses which include decades prior to the 1960's since old ship records or primitive instruments may reflect not a change in climate but rather a change in measurement capabilities or variations in techniques for capturing data.

Certain adjustments to daily data will remain essential since the 1960's, due to inconsistencies in recording zero measurements, or the closing and opening of weather stations, for instance. Changes in data quality have been especially drastic since 1982 as a range of improvements were implemented for achieving more complete, more consistent records. Some of the prominent data changes are presented in summary in the 'Results and Discussion' section.

## 2.7 Actuarial Analysis

Weather is no stranger to the insurance industry; policies insuring ships against storms and other causes of sinkage were first written Before Christ. Modeling weather has become a standard only since Hurricane Andrew, and still catastrophe simulations are proprietary which limits discussion beyond what little the model designers and their clients wish to share. The duration of property insurance policies rarely exceeds one year, so insurers can adjust premiums in response to gradual, long term climate mechanisms and may not need to discern source changes. Primary consideration might be given to ENSO phases, which can be predicted sometimes six months in advance. Other short-horizon climate disruptors may possibly receive some attention. Yet, with growing concern over nature's destructive forces, the role of weather and risk experts may need to be updated to include more than the offering of near-term insurance policies.

Actuaries possess refined comprehension of the messages raveled inside vast sets of data. The need to measure economic costs of calamities has given actuaries a uniquely precise viewpoint of risk assessment. Actuarial science can bring advancements to climate analysis in such areas as credibility and outliers, treatment of sparse data, recognition of interactions, removal of double-counting, identification of noise signals, normalization, trending and pattern searching. Actuaries have placed greater focus on mathematical aspects of storm losses and are far more rigorous in these numerical areas than the other sciences. The treatment of catastrophic weather loss in models combines the skills of the actuary with the atmospheric scientist, together but separately, in a limited market. Techniques in weather and catastrophe may be progress to apply financial and actuarial expertise directly, along with the distinct qualities of physical sciences.

## 3. RESULTS AND DISCUSSION

Results are given from output of the code provided in the appendix, and serve as examples of the much wider range of information the meteorological data sets can provide.

### 3.1 Data Completeness

Stations open and close over time, with changes of location; differences in elevation and surroundings impact measurements. While precipitation (PRCP) has been recorded at over 56,000 stations in the United States and Canada since 1960, fewer than six percent of these stations contain data for 30 base years and the subsequent 27 years for comparison.

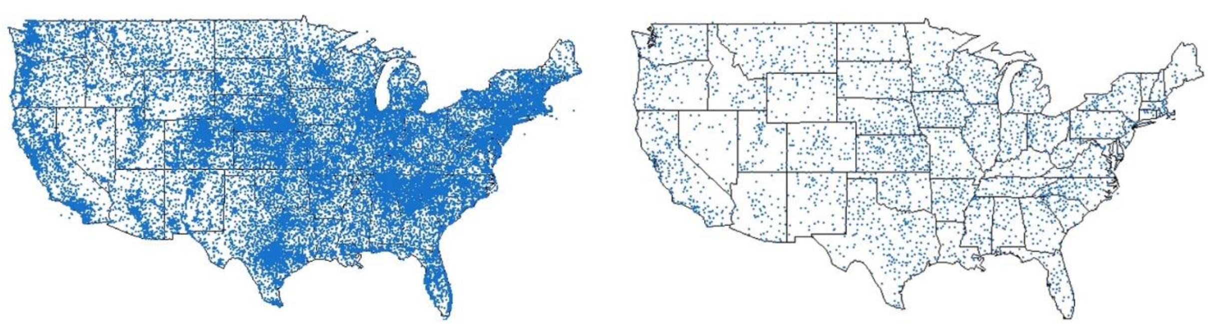


**Figure 7.** Precipitation (rainfall) records have been recorded at over 48,000 stations in the United States since 1960 (left figure). Only 6.4% of these stations records include some data in all 57 years from 1961 to 2017 (right figure); however, over 26% of the yearly precipitation data for these decades was recorded at these long-operating stations.

The GHCN-D data is fraught with missing records. Beginning in 1982, an existing notation became commonly utilized to indicate a blank that had been assumed zero, for quantity measures such as rainfall. The number of identifiable missing records jumped in 1982, and new initiatives were taken so that record completion has improved since then. The practice of assuming zero records was phased out by the end of 2010. Prior to 1982, blanks that were assumed zero cannot be identified, so while the data appears more complete for older years, in reality, the zero records are unreliable.

The National Centers for Environmental Information (NCEI) of the NOAA also provides monthly GHCN data summaries of weather elements (GHCN-M), to which 'homogeneity adjustments' have been made [www.ncdc.noaa.gov/ghcnm]. The online data source includes reference materials describing adjustments that are called for by the raw daily data records.

**Table 4. Change in assumed zeros**. In 1982, GHCN-D missing records appeared to increase only because blanks became identifiable by notation; subsequently completeness has improved.

Missing Records as a % of days of year

| Year | PRCP | SNOW | SNWD | TMAX | TMIN |
|---|---|---|---|---|---|
| 1960 | 2.8% | 4.0% | 6.2% | 2.6% | 2.7% |
| 1961 | 3.2% | 5.4% | 8.5% | 3.2% | 3.3% |
| 1962 | 3.2% | 5.5% | 8.0% | 3.3% | 3.3% |
| … | | | | | |
| 1979 | 5.7% | 8.5% | 11.2% | 7.0% | 6.8% |
| 1980 | 5.4% | 8.8% | 11.5% | 6.8% | 6.8% |
| 1981 | 4.1% | 6.9% | 8.6% | 5.8% | 5.8% |
| 1982 | 31.4% | 82.6% | 78.6% | 4.7% | 4.5% |
| 1983 | 31.3% | 83.0% | 78.9% | 4.2% | 4.1% |
| 1984 | 32.0% | 83.5% | 77.7% | 4.6% | 4.6% |
| … | | | | | |
| 2015 | 21.1% | 41.7% | 44.6% | 4.5% | 4.6% |
| 2016 | 22.0% | 39.5% | 41.3% | 4.7% | 4.8% |
| 2017 | 19.4% | 34.4% | 32.7% | 4.4% | 4.4% |

Zero observations* as a % of observations

| Year | PRCP | SNOW | SNWD | TMAX | TMIN |
|---|---|---|---|---|---|
| 1960 | 75% | 96% | 90% | 9% | 34% |
| 1961 | 73% | 97% | 92% | 7% | 33% |
| 1962 | 74% | 97% | 91% | 8% | 33% |
| … | | | | | |
| 1979 | 73% | 96% | 90% | 10% | 35% |
| 1980 | 75% | 97% | 92% | 8% | 34% |
| 1981 | 74% | 97% | 94% | 7% | 33% |
| 1982 | 60% | 78% | 58% | 9% | 34% |
| 1983 | 60% | 79% | 62% | 9% | 33% |
| 1984 | 61% | 80% | 63% | 8% | 34% |
| … | | | | | |
| 2015 | 68% | 96% | 79% | 8% | 31% |
| 2016 | 69% | 96% | 82% | 8% | 31% |
| 2017 | 68% | 96% | 82% | 8% | 32% |

* For temperatures, 'zero observations' are counts at or below zero degrees Celsius (freezing temperatures).

It might be expected that rainfall (snowfall) might not be recorded reliably during extremely dry (hot) weather. For snowfall in Minnesota, missing records average 45% for summer months for which all records are zeros, but still over 25% of records are missing in snowy winter months.

**Table 5. United States Precipitation Records.** The percentage of daily records each year seems to be falling while actually data quality is improving. Prior to 1982, blank records were assumed zero but most lacked identifying notation.

| Year | % days with record | % days record missing | % days blank assumed zero | % days zero | % records zero | blank assumed zero | count of stations |
|---|---|---|---|---|---|---|---|
| 1961 | 95.1% | 3.2% | 0.1% | 69.7% | 73.3% | 2,343 | 9,704 |
| 1962 | 92.2% | 3.2% | 0.1% | 68.7% | 74.5% | 2,291 | 9,757 |
| 1963 | 94.6% | 3.3% | 0.1% | 72.3% | 76.4% | 2,762 | 9,479 |
| ... | | | | | | | |
| 1979 | 92.7% | 5.7% | 0.1% | 67.6% | 72.9% | 2,447 | 8,527 |
| 1980 | 92.1% | 5.4% | 0.1% | 69.2% | 75.1% | 2,477 | 8,650 |
| 1981 | 94.2% | 4.1% | 0.1% | 69.6% | 73.9% | 2,658 | 8,690 |
| 1982 | 67.3% | 31.4% | 25.8% | 40.2% | 59.7% | 817,610 | 8,673 |
| 1983 | 67.7% | 31.3% | 26.4% | 40.4% | 59.7% | 833,783 | 8,646 |
| 1984 | 67.0% | 32.0% | 27.0% | 40.9% | 61.0% | 851,171 | 8,608 |
| ... | | | | | | | |
| 2008 | 69.3% | 20.0% | 5.8% | 47.5% | 68.5% | 428,242 | 20,058 |
| 2009 | 70.6% | 20.1% | 3.9% | 47.3% | 67.0% | 323,319 | 22,537 |
| 2010 | 71.9% | 21.1% | 3.1% | 50.1% | 69.7% | 268,530 | 23,497 |
| 2011 | 71.8% | 21.1% | 0.0% | 50.0% | 69.6% | | 24,411 |
| 2012 | 72.2% | 20.4% | 0.0% | 51.9% | 72.0% | | 25,516 |
| 2013 | 71.9% | 20.9% | 0.0% | 49.8% | 69.3% | | 26,427 |
| 2014 | 71.7% | 21.4% | 0.0% | 49.3% | 68.9% | | 26,366 |
| 2015 | 72.7% | 21.1% | 0.0% | 49.2% | 67.6% | | 26,017 |
| 2016 | 73.9% | 22.0% | 0.0% | 51.1% | 69.2% | | 24,381 |
| 2017 | 72.5% | 19.4% | 0.0% | 49.5% | 68.3% | | 25,605 |

The change in missing records is explained Dr. Matt Menne, the creator of the GHCN-Daily meteorological databank at the NOAA's National Centers for Environmental Information (NCEI):

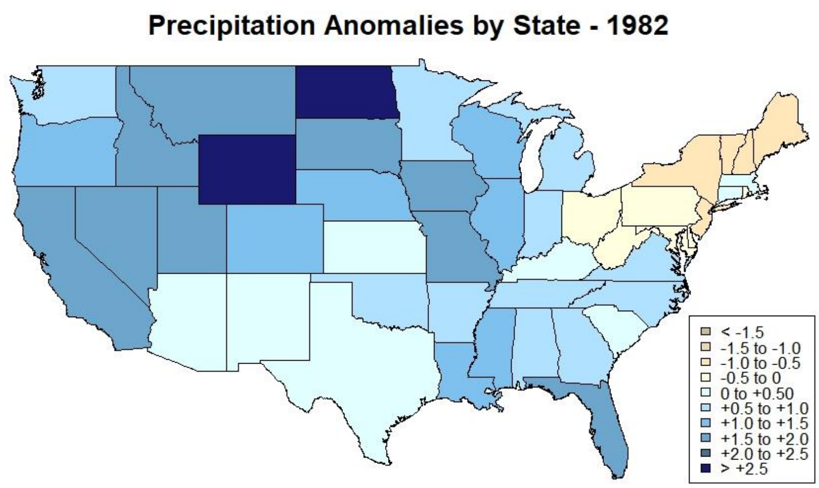
> "Many volunteer observers, especially in the more historic past, have not consistently recorded zeros each day when no rain was observed and rather would often leave the day blank in such cases. Because zeros have so often been left blank on reporting forms, NCEI used to more or less routinely assign a zero value to daily precipitation totals that were left blank. These added zeros were intended to be accompanied by a flag noting that the value "was missing but presumed zero" so that they could be distinguished from days when the observer noted a zero. However, the practice of assuming zeros for blanks was discontinued after 2010 when we moved to a new ingest and processing system for daily data, largely because the accuracy of assuming a zero for blanks could not be assessed very well. In addition, volunteer observers were rapidly transitioning to electronic reporting around the same time and are now prompted somewhat by the new electric entry system as to whether a missing value was really meant to be reported as a zero."

## 3.2 Choropleth Maps

Choropleth maps are color coded ranges that allow immediate visual interpretation. R contains numerous packages that will produce a choropleth map, although most are designed for quantity measures and lack flexibility for other purposes. The example choropleths plot anomalies centered at zero, which is straightforward to code in the package 'ggplot2' but may be more cumbersome to produce with other packages. The package 'ggplot2' has an advantage of being compatible with 'fiftystater' that includes insets of Alaska and Hawaii.

The first choropleth example is created from scratch in package 'maps' and provides code that allows for a high degree of customization. A drawback of this package is the lack of insets for Alaska and Hawaii, although these states can still be mapped separately.

The code allows for a year to be selected, which is compared against the base climate period (1961-1990). The base period average and standard deviation are calculated for each state separately. The choropleth shows the number of deviations upward or downward from the base average.



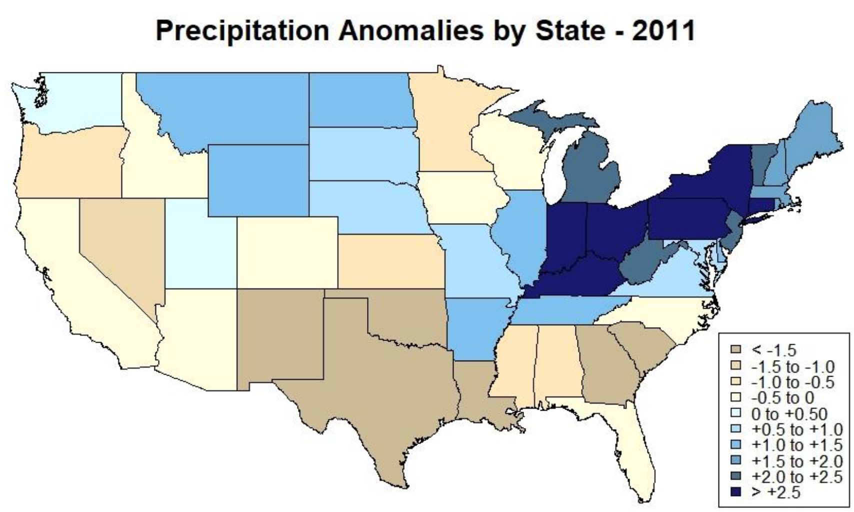
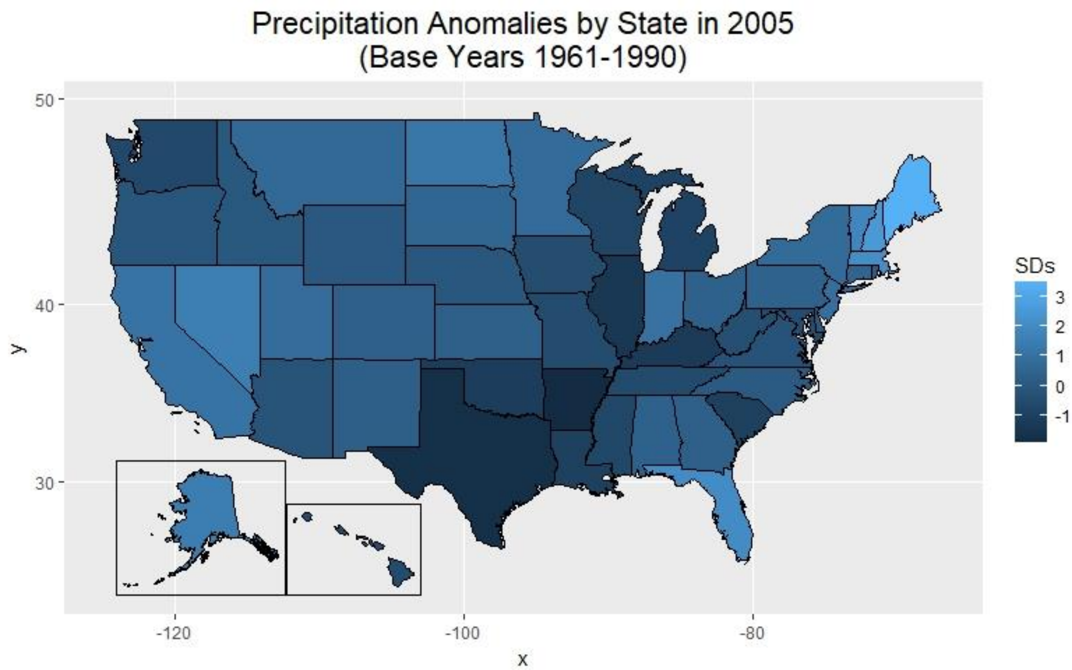




**Figure 8.** Choropleth maps produced from scratch using package 'maps' for an El Niño event year 1982 (top) in contrast to a La Niña event year 2011 (bottom).



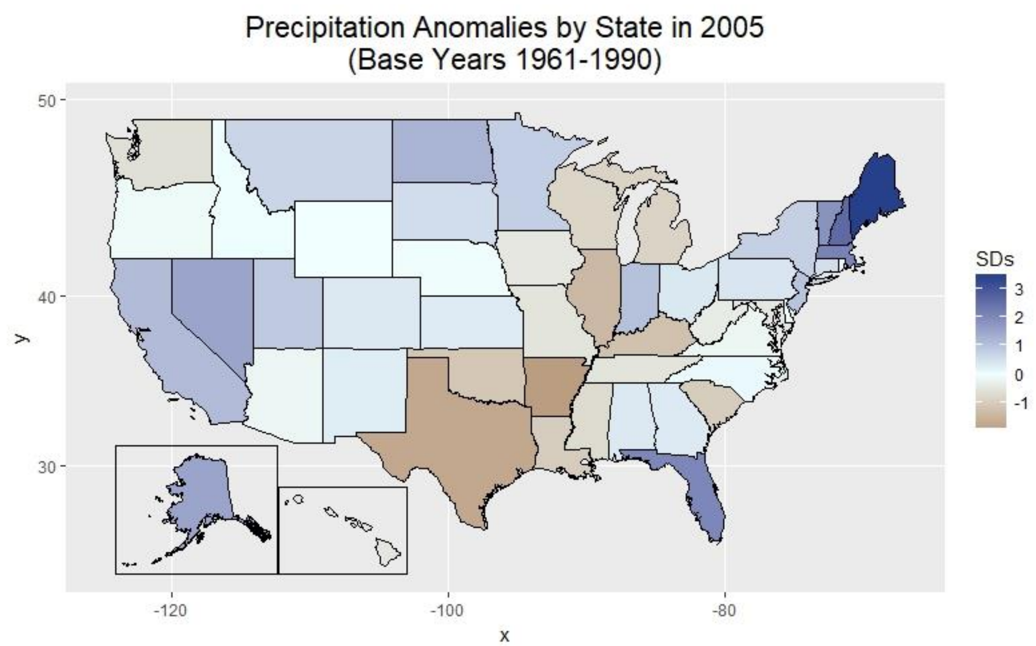




**Figure 9.** Choropleth map produced by package 'ggplot2' with package 'fiftystater' insets, using the base color scheme (top); and using a custom color scheme with a midpoint specified at zero anomaly (bottom).

## 3.3 Plots of Elements vs. Indices

Florida winter precipitation (PRCP) is chosen as an example region from the NOAA Jetstream maps, which indicate wet and cool conditions are expected during El Niño phases, dry and warm during La Niña. Several ENSO indices and time periods are selected to plot against the average daily recorded rainfall. Only stations have been included with some data in all 57 years (1961-2017); data completeness by month has not been checked. No adjustment has been made for assumed zero entries prior to 1982 which lack notation as blank records. The plots assign a shape to distinguish points in the two decades before 1982 which could adjust upwards due to an over prevalence of zeros.

For the Niño indices, there does not appear to be a strong relationship. For the MEI, the correlation with Florida rainfall appears convincing from January to March, but not in December. By this example, the choice of index would appear critical for identifying the specific characteristics of ENSO that impacts the region. If a loss threshold has been established for Florida rainfall, then a relationship between the MEI and the threshold might cause an insurer to consider ENSO phases in its loss history and realign expectations for the future.

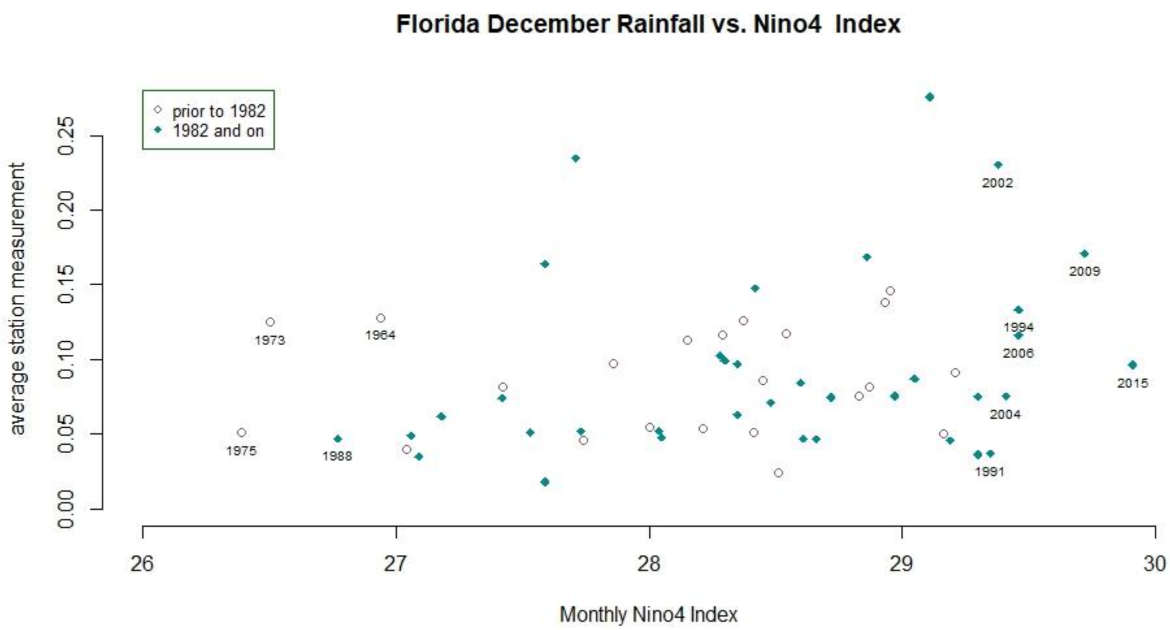



**Figure 10.** Florida rainfall (PRCP) in December plotted against the Niño 4 Index does not reveal a distinct pattern.

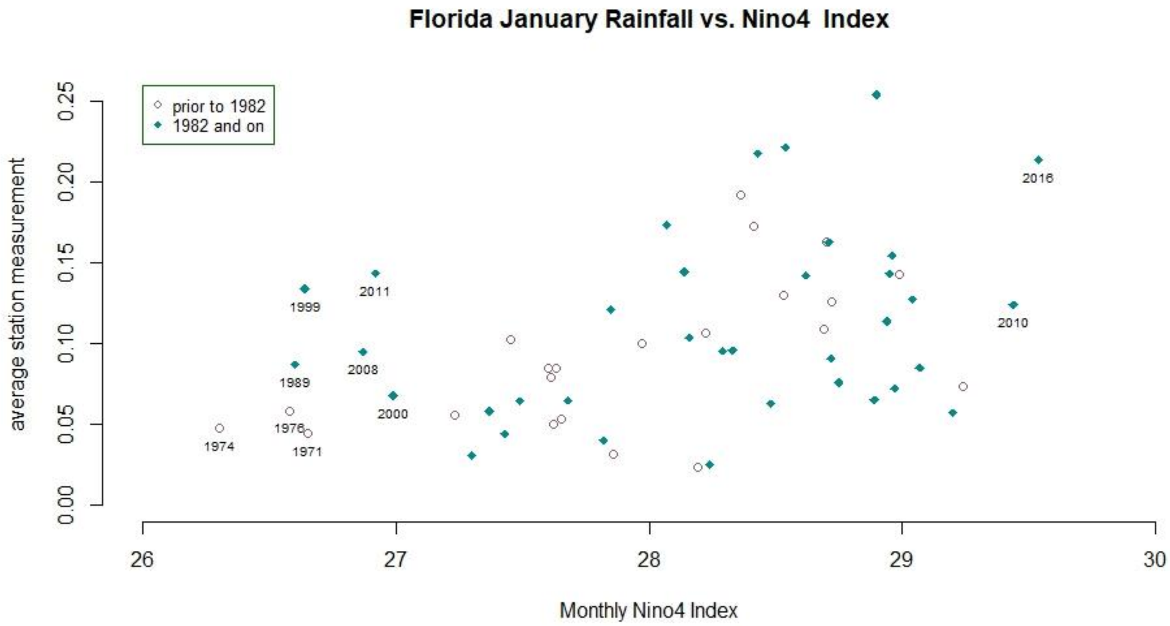



**Figure 11.** Florida rainfall (PRCP) in January plotted against the Niño 4 Index does not reveal a strong phase relationship.

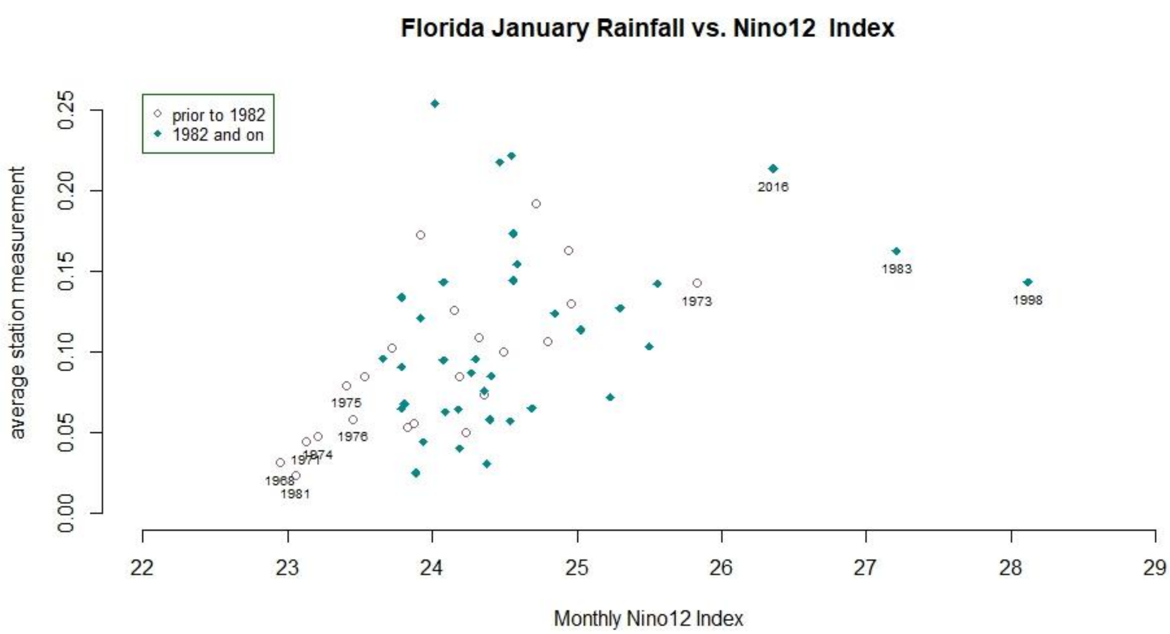



**Figure 12.** Florida rainfall (PRCP) in January plotted against the Niño 1+2 Index might reveal a slight phase relationship.

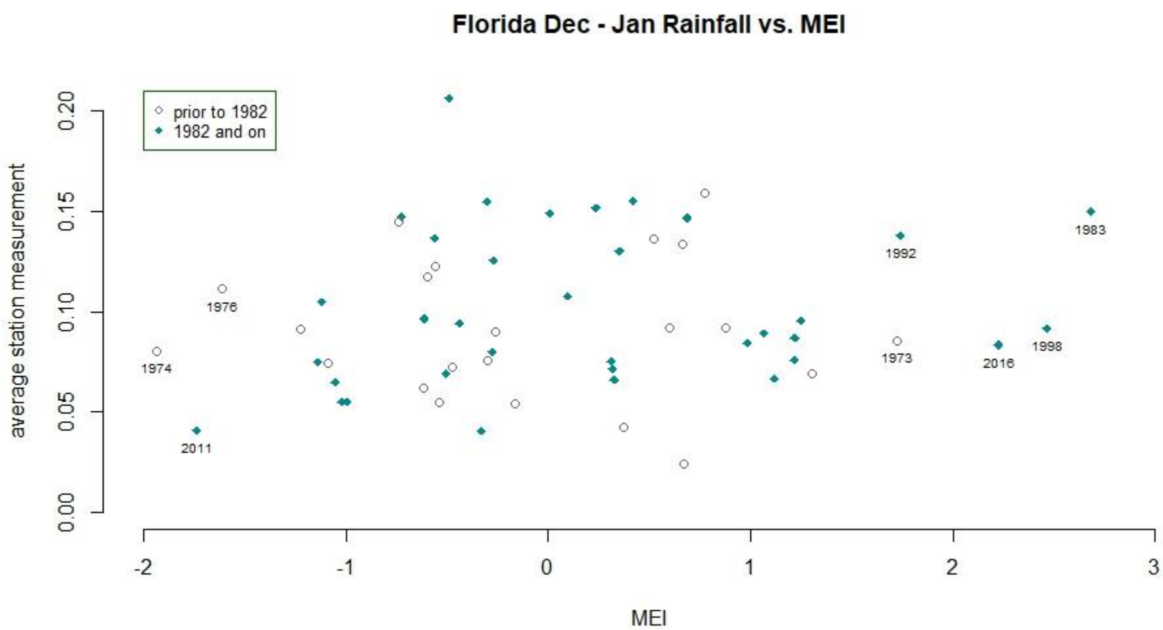



**Figure 13.** Florida rainfall (PRCP) in December and January plotted against the Multivariate ENSO Index (MEI) appears random.

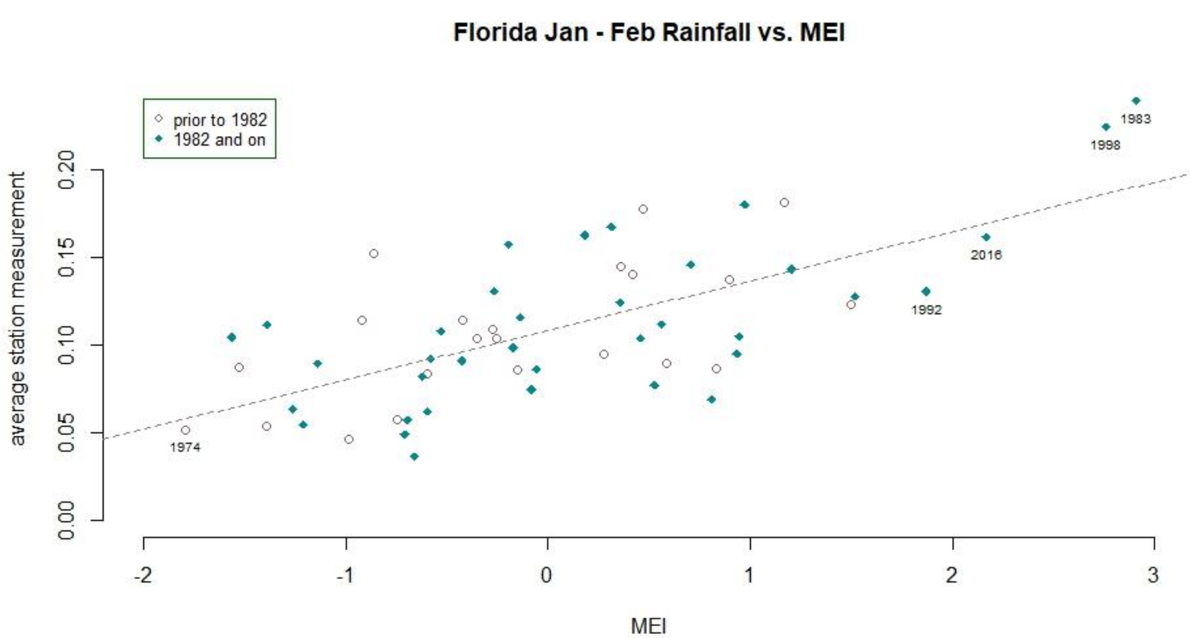



**Figure 14.** Florida rainfall (PRCP) in January and February plotted against the Multivariate ENSO Index (MEI) reveals a positive correlation.

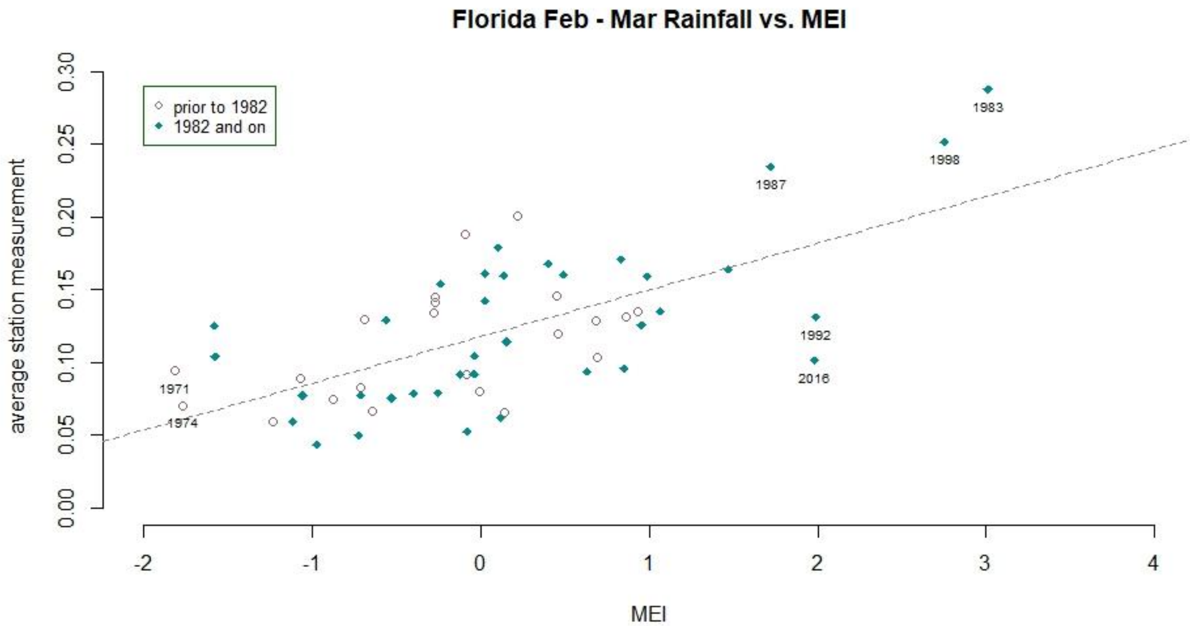



**Figure 15.** Florida rainfall (PRCP) in February and March plotted against the Multivariate ENSO Index (MEI) reveals a positive correlation.

## 4. CONCLUSION

With an uncertain future of weather extremes, one might only expect a deluge in climate stances. A detailed raw data source for weather records in GHNC-D can bring some tangibility, at least to the past, to establish a more concrete understanding of the elusive phenomenon of weather. A revised viewpoint would neither presume upward trends in storm losses nor simply level losses to present conditions. Instead the physical process of heat and motion in cycles and patterns, on many scales, might link weather to losses through thresholds. A closer look at distributions and shifts in weather occurring near damageability thresholds might allow losses to definitively enter the climate formula.

If human activity drives any part of climate change, the next technological advancements might be designed to evaluate and financially prepare for the outcome. The objective is to not only use the newest tools to the greatest advantage, but to continually expand our capabilities towards progress, which may include contributions toward an accurate, consistent bank of data with enough stability to distinguish amplitudes, durations and interactions inherent in natural cyclical 'climate change.'

The growing attention to climate as it affects insurance loss may be a calling for actuaries to uncover the hidden message of the meteorological files. The trends in technologies may finally bring the sophisticated topic of climate "down to earth."

**Acknowledgement**

Many thanks for numerous quality resources provided by the National Oceanic and Atmospheric Administration and to the diverse reviewers including Alp Can FCAS MAAA (actuarial), Dr. Matt Menne (GHCN-D data), Jo Himes (oceanography) and AT Adamack (code).

**Supplementary Material**

Code and code description.

# 5. REFERENCES

*About Climate Normals*. National Oceanic and Atmospheric Administration (NOAA). National Weather Service. Retrieved from www.weather.gov/grr/climatenormals

Ahrens, C. Donald. (2013) *Meteorology Today: An Introduction to Weather, Climate, and the Environment*. Brooks/Cole, Cengage Learning.

Chang, Winston. (2013) *R Graphics Cookbook: Practical Recipes for Visualizing Data*. O'Reilly Media, Inc.

*Climate Indices: Monthly Atmospheric and Ocean Time Series.* National Oceanic and Atmospheric Administration (NOAA). Earth System Research Library. Physical Sciences Division. Retrieved from www.esrl.noaa.gov/psd/data/climateindices/list/

*Download Climate Timeseries*. National Oceanic and Atmospheric Administration (NOAA). Global Climate Observing System (GOCS). Atmospheric Observation Panel for Climate (AOPC)/Ocean Observations Panel for Climate (OOPC), Working Group on Surface Pressure. Retrieved from www.esrl.noaa.gov/psd/gcos_wgsp/Timeseries

*Earth's Rotation*. Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Earth%27s_rotation/

*Global Historical Climate Network Daily - Methods.* National Oceanic and Atmospheric Administration (NOAA). National Centers for Environmental Information (NCEI). Retrieved from www.ncdc.noaa.gov/ghcn-daily-methods

*GHCN Monthly.* National Oceanic and Atmospheric Administration (NOAA). National Centers for Environmental Information (NCEI). Retrieved from www.ncdc.noaa.gov/ghcnm/

*ICAT Damage Estimator*. Retrieved from www.icatdamageestimator.com

*List of Costliest Atlantic Hurricanes*. Wikipedia. Retrieved from https://en.wikipedia.org/wiki/List_of_costliest_Atlantic_hurricanes

*Monthly Atmospheric and SST Indices.* National Ocean and Atmospheric Administration (NOAA). National Weather Service. Retrieved from www.cpc.ncep.noaa.gov/data/indices

Orlowski, J. (Director) 2012 *Chasing Ice* [Documentary, Biography] United States: Diamond Docs.

Roger A. Pielke, Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Roger_A._Pielke

Rohli, Robert V. and Vega, Anthony J. (2012) *Climatology*, Second Edition. Jones & Bartlett Learning.

Sarachik, Edward S. and Cane, Mark A. (2010) *The El Niño Southern Oscillation Phenomenon*. Cambridge University Press.

Taylor, Arnold. (2011) *The Dance of Air and Sea: How Oceans, Weather, and Life Link Together*. Oxford University Press.

*The Comprehensive R Archive Network*. The R Foundation for Statistical Computing. Retrieved from https://cran.r-project.org/

*Tropical Weather*: *the El Niño Southern Oscillation (ENSO)*. National Oceanic and Atmospheric Administration (NOAA). National Weather Service. JetStream - An Online School for Weather. Retrieved from www.srh.noaa.gov/jetstream/tropics/enso_impacts.html

**Gwendolyn Anderson ACAS MAAA** is an actuary who grew up in a solar (glass) house beside the epicenter of the Loma Prieta earthquake and later achieved an international position in the New York City financial district one month before the 9/11/01 terrorist attacks. Her background is primarily commercial lines pricing, and includes Atlantic hurricanes and financial and predictive models. Off hours she might be found explaining policy coverage in Spanish at a tornado site or exploring earth-friendly technologies at BioGrasse in France. She recently earned a master's degree in financial mathematics focusing on climatology and meteorology, and is especially interested in areas of perplexing logic, unusual probabilities and high risk.

# Meteorology for Actuaries – Part 2
## Climate and the El Niño Southern Oscillation
# Code and Code Description
(February 2018)



### 5.0 Set up in R

To begin, copy and paste the code into an *.R script file. The code follows the descriptions. If R GUI and R Studio have not yet been installed, instructional videos are available on youtube. After copying code into the *.R script, single quotes may need to be replaced with properly formatted quotes (use <ctrl>-f to find and replace single quotes.) Before running the code, directory paths must be specified and inputs copied into *.csv files.

### Directories and Inputs

The paths to three directories are to be specified in the code where R can locate the initial input files and write output files. The input files for this example will be in *.csv (Comma Delimited) and need to be saved to the directory folders named in the code. The files listed below with the directories are the files to be used in the code examples. The code can be modified to run fewer or more years of zipped *.gz files or to read different base inputs. If expanding years of input, be aware that daily data figures are unadjusted and years before 1960 will be subject to inconsistencies in measurements. The files saved in the first two directories (I) dirzip and (II) dirbase provide the inputs to produce subsets and summaries, which are written out as more accessible *.csv files to (III) diroutput. The output files will be accessed again to run subsequent code much more efficiently than by unzipping cumbersome *.gz files.

The three directories and contents will be as follows:

```
(I)    dirzip <- "C:/…/WeatherZip"
                                   1960.csv.gz
                                   1961.csv.gz
                                   …
                                   2016.csv.gz
                                   2017.csv.gz
(II)   dirbase <- "C:/…/WeatherBase"
                                   BEST1mo.csv
                                   CostlyStorms.csv
                                   EQSOI.csv
                                   ghcnd-inventory.csv
                                   ghcnd-stations.csv
                                   ghcnd-states.csv
                                   MEI.csv
                                   NinoMonthly.csv
                                   NinoWeekly.csv
                                   ONI.csv
                                   SOI_Anom.csv
                                   TNI.csv
(III)  diroutput <- "C:/…/WeatherData"
```

The first folder (I) dirzip contains the zipped files daily data, which are downloaded from the NOAA GHCN-Daily at this website address:

ftp://ftp.ncdc.noaa.gov/pub/data/ghcn/daily/

```
by_year/               folder of zipped files of daily data by year to download
readme.txt             detailed descriptions of variables and their values
ghcnd-stations.txt     StationID, name, coordinates, elevation, state/province abbreviation
ghcnd-countries.txt    two-character GHCN country and territory codes, and names
ghcnd-states.txt       two-character US states and territories, Canadian provinces
ghcnd-inventory.txt    StationID, coordinates, station start and end years by element
```

Although text files can be read by R, it is more reliable overall to copy and parse the data into excel and save as *.csv files. The example code runs data for the US and Canada ('CA'). The file ghcnd-countries.txt gives two-character country codes and country names that can be used as inputs to modify the example. The file ghcnd-inventory.txt provides basic ranges of years during which stations have recorded data, by weather element; this inventory list is longer than ghcnd-stations.txt which lists each station once only. The code will summarize greater station detail from the weather records to assist in selecting consistent data across years. Note that the zip code field in ghcnd-stations.txt is missing entries so it would not serve for mapping stations to counties.

The fields 'Open' and 'Close' in the ghcnd-inventory.txt file were included in the example code at a later time so as not to be shown in the sample outputs of this paper.

Table 1. Sample output from ghcnd-stations.txt saved as *.csv and read by R as a data table.

| | StationID | lat | lon | elev | St | Name | GSNFlag | zip |
|---|---|---|---|---|---|---|---|---|
| 1: | ACW00011604 | 17.1167 | -61.7833 | 10.1 | NA | ST JOHNS COOLIDGE FLD | NA | NA |
| 2: | ACW00011647 | 17.1333 | -61.7833 | 19.2 | NA | ST JOHNS | NA | NA |
| 3: | AE000041196 | 25.333 | 55.517 | 34 | NA | SHARJAH INTER. AIRP | NA | 41196 |
| 4: | AEM00041194 | 25.255 | 55.364 | 10.4 | NA | DUBAI INTL | NA | 41194 |
| 5: | AEM00041217 | 24.433 | 54.651 | 26.8 | NA | ABU DHABI INTL | NA | 41217 |
| --- | | | | | | | | |
| *: | ZI000067969 | -21.05 | 29.367 | 861 | NA | WEST NICHOLSON | NA | 67969 |
| *: | ZI000067975 | -20.067 | 30.867 | 1095 | NA | MASVINGO | NA | 67975 |
| *: | ZI000067977 | -21.017 | 31.583 | 430 | NA | BUFFALO RANGE | NA | 67977 |
| *: | ZI000067983 | -20.2 | 32.616 | 1132 | NA | CHIPINGE | NA | 67983 |
| *: | ZI000067991 | -22.217 | 30 | 457 | NA | BEITBRIDGE | NA | 67991 |

* column numbers not shown (104122 – 104126)

| | |
|---|---|
| StationID | station identification number |
| lat | latitude coordinate of station location |
| lon | longitude coordinate of station location |
| elev | elevation of station location |
| St | state or province two-character abbreviation |
| Name | station name |
| GSNFlag | (see readme.txt for details) |
| zip | zip code of station location |

Table 2. Sample output from ghcnd-inventory.txt saved as *.csv and read by R as a data table.

| | StationID | lat | lon | elem | Open | Close |
|---|---|---|---|---|---|---|
| 1: | ACW00011604 | 17.1167 | -61.7833 | TMAX | 1949 | 1949 |
| 2: | ACW00011604 | 17.1167 | -61.7833 | TMIN | 1949 | 1949 |
| 3: | ACW00011604 | 17.1167 | -61.7833 | PRCP | 1949 | 1949 |
| 4: | ACW00011604 | 17.1167 | -61.7833 | SNOW | 1949 | 1949 |
| 5: | ACW00011604 | 17.1167 | -61.7833 | SNWD | 1949 | 1949 |
| --- | | | | | | |
| 596841: | ZI000067983 | -20.2 | 32.616 | PRCP | 1951 | 2017 |
| 596842: | ZI000067983 | -20.2 | 32.616 | TAVG | 1962 | 2017 |
| 596843: | ZI000067991 | -22.217 | 30 | TMAX | 1951 | 1990 |
| 596844: | ZI000067991 | -22.217 | 30 | TMIN | 1951 | 1990 |
| 596845: | ZI000067991 | -22.217 | 30 | PRCP | 1951 | 1990 |

| | |
|---|---|
| Open | first year the station recorded data for the weather element specified |
| Close | final year the station recorded data for the weather element specified |

Table 3. Sample output from ghcnd-states.txt (left) and ghcnd-countries.txt (right) saved as *.csv files and read by R as data tables.

| | St | Name | \|\| | | loc | CountryName |
|---|---|---|---|---|---|---|
| 1: | AB | ALBERTA | \|\| | 1: | AC | Antigua and Barbuda |
| 2: | AK | ALASKA | \|\| | 2: | AE | United Arab Emirates |
| 3: | AL | ALABAMA | \|\| | 3: | AF | Afghanistan |
| 4: | AR | ARKANSAS | \|\| | 4: | AG | Algeria |
| 5: | AS | AMERICAN SAMOA | \|\| | 5: | AJ | Azerbaijan |
| --- | | | \|\| | --- | | |

```
   70:   WA     WASHINGTON           ||       214:   WI      Western Sahara
   71:   WI     WISCONSIN            ||       215:   WQ      Wake Island [United States]
   72:   WV     WEST VIRGINIA        ||       216:   WZ      Swaziland
   73:   WY     WYOMING              ||       217:   ZA      Zambia
   74:   YT     YUKON TERRITORY      ||       218:   ZI      Zimbabwe
-------------------------------------------------------------------------------------------
loc           two-character abbreviation for country or territory
```

The file IndexMonthly.csv is created by code, combining various monthly indices that have been accessed separately from online sources and saved into *.csv files. ENSO indices used in the sample code can be copied from these sources:

| Website Address | | Indices (format: Wide or Long) | from |
|---|---|---|---|
| (I) | www.cpc.ncep.noaa.gov/data/indices/ | Niño, ONI (L); SOI, EQSOI (W) | 1950's |
| | | Niño Weekly (L) | 1990's |
| (II) | www.esrl.noaa.gov/psd/gcos_wgsp/Timeseries/ | Niño, SOI (W) | 1870's |
| (III) | www.esrl.noaa.gov/psd/data/climateindices/list/ | Niño, ONI SOI, TNI, BEST, MEI (W) | 1950's |
| (IV) | www.esrl.noaa.gov/psd/enso/mei/table.html | MEI (W) | 1950's |

The first online resource (I) includes all of the monthly Niño indices and anomalies in one file, dating from the 1950's (ERSST monthly). Weekly Niño indices and anomalies are also available in one file, although only from the 1990's (OISST weekly). The ONI is given in a separate file (ERSST seasonal) also from the 1950's. These indices are given in a "long" format, indicated above as (L), where months are stacked in one column. The SOI and EQSOI are each given in separate files from the 1950's in a "wide" format (W) where each month is in a separate column. In R code, the "wide" format can be converted to "long," or vice versa, using package 'tidyr' functions (i.e. gather() and spread()).

The second online resource (II) provides a number of climate indices, including each of the Niño indices given separately in "wide" format from 1870, and each anomaly separately also. The SOI is similarly given in "wide" format monthly back to 1866. The older years may be of limited value for comparison against inconsistent weather data. The third online resource (III) also contains numerous climate indices, and is a source for the Trans-Niño Index (TNI) in wide format from the 1950's. The fourth resource (IV) is the direct site for the Multivariate ENSO Index (MEI).

## Running Code

To run one or multiple lines of code, highlight the code and press <ctrl>-r. To run one line of code, alternatively, place the cursor at the line and press <ctrl>-<enter>. To run a 'for loop' highlight the entire loop from 'for' to the end bracket '}' and press <ctrl>-r. Comment lines begin with '#' and will not run.

## Packages

```
data.table    functions run faster than base R code
                   rbindlist() to combine years of weather data frames from a list
                   setnames() to update column headers
```

```
tidyverse      a set of packages for organizing data
                   package 'readr' to unzip *.gz files
                   package 'dplyr' to calculate statistics
                   package 'ggplot2' to map choropleths
               package 'tidyr' functions to convert formats between wide and long
lubridate      functions to calculate number of days in month, leap years, etc.
```

Tutorials are available online for instruction on utilizing the data.table functions advantageously.

### Common Errors

Because the daily data is voluminous, errors encountered running code may involve space and memory. "Error: cannot allocate vector of size _ Mb" may occur if many large data sets are stored in the environment. The command ls() can be used to view current data sets, and rm() can be used to remove a data set specified within the parentheses. To free memory, the computer can be completely shut down and restarted without opening programs other than R. If a 'for loop' stops prior to completion, the command ls() can be used to identify the latest data set so the code can be continued from that point inside the brackets; a shorter loop can then be defined based on the remaining years or elements.

## 5.1 Code 1: Weather Daily Loop

```
-----------------------------------------------------------------------------------------------
Table 4. Code 1 Sample Output.  Weather Daily Loop.  Precipitation, US and Canada.

   StationID   date     elem VAL  MFlag QFlag SFlag Time loc  year month monthday VAL_US
1: CA001010720 19600101 PRCP 0     -     -     C     -   CA   1960  1     101      0
2: CA001010720 19600102 PRCP 25    -     -     C     -   CA   1960  1     102      0.098425
3: CA001010720 19600103 PRCP 0     T     -     C     -   CA   1960  1     103      0
4: CA001010720 19600104 PRCP 41    -     -     C     -   CA   1960  1     104      0.161417
5: CA001010720 19600105 PRCP 257   -     -     C     -   CA   1960  1     105      1.011811
---
*  USW00094967 19601227 PRCP 0     T     -     0     -   US   1960  12    1227     0
*  USW00094967 19601228 PRCP 0     -     -     0     -   US   1960  12    1228     0
*  USW00094967 19601229 PRCP 0     T     -     0     -   US   1960  12    1229     0
*  USW00094967 19601230 PRCP 0     -     -     0     -   US   1960  12    1230     0
*  USW00094967 19601231 PRCP 5     -     -     0     -   US   1960  12    1231     0.019685
-----------------------------------------------------------------------------------------------
*  column numbers not shown (3982198 – 3982201)

date           yyyymmdd format
VAL            record in metric system units according to weather element, given in Table 5
MFlag          includes notation 'P' for blank records assumed zero
QFlag          (See readme.txt for details)
SFlag          (See readme.txt for details)
Time           (See readme.txt for details)
year           field created from date
month          field created from date
monthday       field created from date
VAL_US         conversion of VAL field to US Imperial system units, given in Table 5
```

Code 1 is a double loop that unzips the massive meteorological data files containing the daily station detail of weather element measurements. Since unzipping requires the most run time, for each year unzipped the code loops through weather elements to write out to separate *.csv files by element. Wind data is sparse so a few of the GHNC elements are combined in one output file as 'WIND' as a collective term, not a GHNC element. The number of *.csv files to be written out equals the number of years selected in the outer loop times the number of elements selected.

Table 5. Weather elements included in sample code.

FIVE CORE ELEMENTS

| Abbr | Element | Unit of Measure | Converted (US) |
|---|---|---|---|
| PRCP | Precipitation | tenths of mm | inches |
| SNOW | Snowfall | mm | inches |
| SNWD | Snow depth | mm | inches |
| TMAX | Maximum temperature | tenths of degrees C | degrees Fahrenheit |
| TMIN | Minimum temperature | tenths of degrees C | degrees Fahrenheit |

WIND elements are coded to include:

| Abbr | Element | Unit of Measure | US |
|---|---|---|---|
| AWND | Average daily wind speed | tenths of meters per second | mph |
| WSF1 | Fastest 1-minute wind speed | tenths of meters per second | mph |
| WSF2 | Fastest 2-minute wind speed | tenths of meters per second | mph |
| WSF5 | Fastest 5-second wind speed | tenths of meters per second | mph |
| WSFG | Peak gust wind speed | tenths of meters per second | mph |
| WSFI | Highest instantaneous wind speed | tenths of meters per second | mph |
| WSFM | Fastest mile wind speed | tenths of meters per second | mph |

To preserve memory resources, time, and storage, few calculations are made while unzipping. Only five columns are added, the location (country/territory) for selection purposes, a few date fields (year, month, month-day), and the U.S. measurement conversion. The sample countries selected are United States and Canada, which are manageable with 8GB memory. Additional countries or territories may need to be selected separately to avoid errors from inadequate memory; while countries with extremely sparse data may needed to be selected in combination so the code will not stop. Additional weather elements that may be selected are listed in the 'readme.txt' file at the GHCN-D site. The 'readme.txt' file provides descriptions for the fields in the data files represented by the data set column names.

The MFlag notation 'P' in the daily records is quite critical as it represents blank records assumed as zero. This notation applies only to quantity elements like rainfall, and not to continuous measures such as temperature. MFlag also has a notation 'T' that R can mistake for a logical (True/False) causing the 'P' notations to be deleted when writing out to a saved file. The code converts empty cells to dashes which avoids losing data in an unintended format conversion.

Code 3 will adjust counts of zero and blank entries based on the MFLAG 'P' notation, but the notation was not widely used before 1982. The data requires adjustments for unidentified blanks assumed zero prior to 1982, and for the improvement in completion of zero records since 1982.

### 5.2 Code 2: Detailed Station Inventories

```
-----------------------------------------------------------------------------------------------
 Table 6.  Code 2 Sample Output. Detailed Station Inventories.  Precipitation, US and Canada.

          StationID     loc    St    lat      lon     elev    elem    mindate     maxdate
 1:       CA001010066   CA     BC    48.867  -123.3      4    PRCP    19840701    19961129
 2:       CA001010235   CA     BC    48.4    -123.5     17    PRCP    19710601    19950430
 3:       CA001010595   CA     BC    48.583  -123.5     85    PRCP    19610208    19801128
 4:       CA001010720   CA     BC    48.5    -124      351    PRCP    19600101    19710831
 5:       CA001010780   CA     BC    48.333  -123.6     12    PRCP    19600101    19660430
 ---
 56621:   USW00094996   US     NE    40.695   -96.85    NA    PRCP    20020622    20171231
 56622:   USW00096404   US     AK    62.737  -141.2     NA    PRCP    20110926    20171231
 56623:   USW00096406   US     AK    64.501  -154.1     NA    PRCP    20140829    20171231
 56624:   USW00096407   US     AK    66.562  -159       NA    PRCP    20150814    20171231
 56625:   USW00096408   US     AK    63.452  -150.9     NA    PRCP    20150820    20171214

          minyear    minmo    maxyear    maxmo     clsdmbeg    clsdmend    clsdinm    clsdbef
 1:         1984     7          1996      11            0           1          1        182
 2:         1971     6          1995       4            0           0          0        151
 3:         1961     2          1980      11            7           2          9         31
 4:         1960     1          1971       8            0           0          0          0
 5:         1960     1          1966       4            0           0          0          0
 ---
 56621:     2002     6          2017      12           21           0         21        151
 56622:     2011     9          2017      12           25           0         25        243
 56623:     2014     8          2017      12           28           0         28        212
 56624:     2015     8          2017      12           13           0         13        212
 56625:     2015     8          2017      12           19          17         36        212

          clsdaft    clsdfulm    bsyrct    rcyrct     bryrct     bsspan     rcspan     brspan
 1:         31         213         7         6          13          7          6         13
 2:         245        396         8         5          13         20          5         25
 3:         31         62          20        0          20         20          0         20
 4:         122        122         11        0          11         11          0         11
 5:         245        245         6         0           6          6          0          6
 ---
 56621:     0          151         0         16         16          0         16         16
 56622:     0          243         0         7           7          0          7          7
 56623:     0          212         0         4           4          0          4          4
 56624:     0          212         0         3           3          0          3          3
 56625:     0          212         0         3           3          0          3          3
-----------------------------------------------------------------------------------------------
StationID     Station identification number from GHCN-D.
loc           two-character country code, the first two digits of the StationID
St            two-character state or province abbreviation where station is located
lat           latitude coordinate of station location
lon           longitude coordinate of station location
elev          elevation of station location
elem          weather element
Open          (not shown) first year of station records from ghcnd-inventory.csv
Close         (not shown) last year of station records from ghcnd-inventory.csv
mindate       earliest date of station record, from years selected to summarize
maxdate       latest date of station record, from years selected to summarize
minyear       year of mindate
minmo         month of mindate
maxyear       year of maxdate
maxmo         month of maxdate
clsdmbeg      count of days the station was not yet operating at beginning of partial month
```

```
clsdmend      count of days the station was no longer operating at end of partial month
clsdinm       total days station did not operate in partial months (clsdmbeg + clsdmend)
clsdbef       count of days in year the station was not yet operating for full months
clsdaft       count of days in year the station no longer was operating for full months
clsdfulm      count of days in year the station was closed for full months (clsdbef + clsdaft)
bsyrct        count of years with observations in the selected base period
rcyrct        count of years with observations in recent years following base period
bryrct        count of years with observations in base and recent years
bsspan        span of base years that fall within the mindate and the maxdate
rcspan        span of years following the base period within the mindate and the maxdate
brspan        span of base and recent years that fall within the mindate and the maxdate
```

Code 2 is an intermediate step to conserve memory. Code 2 reads in the subsets of daily data from Code 1 and produces summaries of station inventories separately for each weather element, for all stations with at least one record in the years of data selected. The focus of the summary is to calculate the minimum and maximum date of record. These dates are compared against the station's opening and closing years given in ghcnd-inventory.txt, to arrive at counts of days the station was not operating in partial months (also full months), for the month (also the year) the station opened or closed. The summary counts years with records for the selected base years and recent years, and also calculates the span of time from the earliest to latest year of record without deducting empty data years. The summary again merges, with data from ghcnd-stations.txt, to list each station's location by coordinates and state. All years from the daily files are combined into one file for each element, so the number of files output is equal to the number of elements selected. If the ghnc-station.txt list is incomplete, the two-character state abbreviation can be found in the StationID for more recent years; but the elevation and coordinates will be missing from final outputs.

## 5.3 Code 3: Initial Year-Month Summaries

```
------------------------------------------------------------------------------------------
 Table 7.  Code 3 Sample output. Initial Year-Month Summary.  Precipitation, US and Canada.

          StationID      loc    elem     year    month     VALm     VALm_US     sumVALsqd_US
 1:       CA001010720    CA     PRCP     1960      1       3497     13.7677165   20.0084785
 2:       CA001010720    CA     PRCP     1960      2       4155     16.3582677   29.8169911
 3:       CA001010720    CA     PRCP     1960      3       3839     15.1141732   24.9078523
 4:       CA001010720    CA     PRCP     1960      4       3486     13.7244094   16.6643623
 5:       CA001010720    CA     PRCP     1960      5       2412      9.496063     6.6623473
 ---
 9001954: USW00096408    US     PRCP     2017      8        800      3.1496063    1.1848844
 9001955: USW00096408    US     PRCP     2017      9        623      2.4527559    0.7029729
 9001956: USW00096408    US     PRCP     2017      10       895      3.523622     1.6202957
 9001957: USW00096408    US     PRCP     2017      11       598      2.3543307    0.6302003
 9001958: USW00096408    US     PRCP     2017      12       211      0.8307087    0.1854269

             VALsqm_US    zerrec     recs    zblank    zerobs    obs    daysinmo  misclsd
 1:         189.5500186     10        31       0         10       31       31        0
 2:         267.5929227     12        29       0         12       29       29        0
 3:         228.4382324     11        31       0         11       31       31        0
 4:         188.3594147      9        30       0          9       30       30        0
 5:          90.1752124      6        31       0          6       31       31        0
 ---
```

```
 9001954:     9.9200198      9       30      0        9     30      31        1
 9001955:     6.0160115     15       30      0       15     30      30        0
 9001956:    12.4159123     15       31      0       15     31      31        0
 9001957:     5.5428731     17       30      0       17     30      30        0
 9001958:     0.6900769      9       14      0        9     14      31       17
-----------------------------------------------------------------------------------------------
VALm          monthly sum of daily VAL
VALm_US       monthly sum of daily VAL_US, VAL converted to U.S. unit of measure.
sumVALsqd_US  sum of squared daily values in U.S. units, to compute daily variances
VALsqm_US     squared monthly values in U.S. units, to computer monthly variances
zerrec        count of records with zero values (quantities); zero and below (temperature)
recs          count of all records
zblank        blank records assumed zero and recorded as zero (for quantity elements)
zerobs        net (below) zero observations, after subtracting blanks assumed zero
obs           net observations, after subtracting blanks assumed zero
daysinmo      count of days in calendar month, representing maximum possible records
misclsd       count of days in month with missing records, or days station not open
```

Code 3 is an intermediate step to conserve memory. Code 3 reads in the subsets of daily data from Code 1 and summarizes monthly data. It also provides important counts of records, for use in selecting stations with adequately complete data or for identifying areas in need of adjustments.

A critical threshold is calculated as variable 'zerobs' which for quantities like rainfall counts zero values relating to drought. For temperatures, the critical threshold 'zerobs' represents freezing temperatures, zero degrees Celsius and below.

A critical data adjustment is made to quantity records through a code calculation, deducting blank records assumed zero ('zblank') from zero records ('zerrec') and records ('recs') to arrive at the adjusted count of observations ('obs') and zero observations ('zerobs'). This calculation does not hold for years before 1982 when the MFlag notation 'P' was rarely used to identify blanks assumed zeros.

No adjustment has been made to the raw daily data records either for blank records assumed zero, or to account for the continued improvement in record completion since 1982. Therefore, the GHNC-D data is in need of adjustments for changes in record keeping over time. No other adjustments are made by the example code, besides converting 'zblank' record counts from zero to blank.

A number of adjustments have been made in the monthly summary GHNC-M, available online [www.ncdc.noaa.gov/ghcnm] with references on the 'homogeneity adjustment' that could be considered when analysis necessitates consistency along with the detail of the raw daily data set.

## 5.4 Code 4: Complete Year and Year-Month Summaries Merged with Station Locations and Inventories

Table 8. Code 4 Sample output (I). Complete Year-Month Summary. Precipitation, US and Canada

| | StationID | loc | St | lat | lon | elev | mindate | maxdate |
|---|---|---|---|---|---|---|---|---|
| 1: | CA001010720 | CA | BC | 48.5 | -124 | 351 | 19600101 | 19710831 |
| 2: | CA001010720 | CA | BC | 48.5 | -124 | 351 | 19600101 | 19710831 |
| 3: | CA001010720 | CA | BC | 48.5 | -124 | 351 | 19600101 | 19710831 |
| 4: | CA001010720 | CA | BC | 48.5 | -124 | 351 | 19600101 | 19710831 |
| 5: | CA001010720 | CA | BC | 48.5 | -124 | 351 | 19600101 | 19710831 |
| 6: | CA001010720 | CA | BC | 48.5 | -124 | 351 | 19600101 | 19710831 |

| | elem | year | month | VALm | VALm_US | sumVALsqd_US | VALsqm_US |
|---|---|---|---|---|---|---|---|
| 1: | PRCP | 1960 | 1 | 3497 | 13.768 | 20.008 | 189.550019 |
| 2: | PRCP | 1960 | 2 | 4155 | 16.358 | 29.817 | 267.592923 |
| 3: | PRCP | 1960 | 3 | 3839 | 15.114 | 24.908 | 228.438232 |
| 4: | PRCP | 1960 | 4 | 3486 | 13.724 | 16.664 | 188.359415 |
| 5: | PRCP | 1960 | 5 | 2412 | 9.4961 | 6.6623 | 90.175212 |
| 6: | PRCP | 1960 | 6 | 535 | 2.1063 | 0.7874 | 4.436496 |

| | zerrec | recs | zblank | zerobs | obs | daysinmo | misinm | clsdinm |
|---|---|---|---|---|---|---|---|---|
| 1: | 10 | 31 | 0 | 10 | 31 | 31 | 0 | 0 |
| 2: | 12 | 29 | 0 | 12 | 29 | 29 | 0 | 0 |
| 3: | 11 | 31 | 0 | 11 | 31 | 31 | 0 | 0 |
| 4: | 9 | 30 | 0 | 9 | 30 | 30 | 0 | 0 |
| 5: | 6 | 31 | 0 | 6 | 31 | 31 | 0 | 0 |
| 6: | 18 | 30 | 0 | 18 | 30 | 30 | 0 | 0 |

clsdinm count of days station closed in partial month, sum (clsdmbeg + clsdmend)
misinm count of missing records in partial month, the difference (misclsd - clsdinm)

Table 9. Code 4 Sample output (II). Complete Year Summary. Precipitation, US and Canada

| | StationID | loc | St | lat | lon | elev | mindate | maxdate |
|---|---|---|---|---|---|---|---|---|
| 1: | CA001010720 | CA | BC | 48.5 | -124 | 351 | 19600101 | 19710831 |
| 2: | CA001010780 | CA | BC | 48.3333 | -123.633 | 12 | 19600101 | 19660430 |
| 3: | CA001010965 | CA | BC | 48.5667 | -123.433 | 91 | 19600801 | 19700630 |
| 4: | CA001011500 | CA | BC | 48.9333 | -123.75 | 75 | 19600101 | 20171231 |
| 5: | CA001011920 | CA | BC | 48.5333 | -123.367 | 37 | 19600101 | 19700331 |
| 6: | CA001012010 | CA | BC | 48.7167 | -123.55 | 1 | 19600101 | 20010311 |

| | elem | year | VALy | VALy_US | sumVALsqd_US | sumVALsqm_US | VALsqy_US |
|---|---|---|---|---|---|---|---|
| 1: | PRCP | 1960 | 33091 | 130.27953 | 198.905868 | 1892.87716 | 16972.7553 |
| 2: | PRCP | 1960 | 9473 | 37.29528 | 24.867707 | 170.58961 | 1390.9376 |
| 3: | PRCP | 1960 | 3842 | 15.12598 | 8.665075 | 63.39429 | 228.7954 |
| 4: | PRCP | 1960 | 11755 | 46.27953 | 39.69544 | 259.06651 | 2141.7947 |
| 5: | PRCP | 1960 | 7778 | 30.62205 | 16.595573 | 107.77906 | 937.7098 |
| 6: | PRCP | 1960 | 9984 | 39.30709 | 29.90204 | 184.49805 | 1545.0471 |

| | zerrec | recs | zblank | zerobs | obs | daysum | daysinyr | monthct |
|---|---|---|---|---|---|---|---|---|
| 1: | 167 | 366 | 0 | 167 | 366 | 366 | 366 | 12 |
| 2: | 208 | 366 | 0 | 208 | 366 | 366 | 366 | 12 |
| 3: | 82 | 148 | 0 | 82 | 148 | 153 | 366 | 5 |
| 4: | 210 | 366 | 0 | 210 | 366 | 366 | 366 | 12 |
| 5: | 223 | 364 | 0 | 223 | 364 | 366 | 366 | 12 |
| 6: | 208 | 366 | 0 | 208 | 366 | 366 | 366 | 12 |

| | clsdinm | clsdfulm | clsdall | misinm | misfulm | misall |
|---|---|---|---|---|---|---|
| 1: | 0 | 0 | 0 | 0 | 0 | 0 |
| 2: | 0 | 0 | 0 | 0 | 0 | 0 |
| 3: | 0 | 213 | 213 | 5 | 0 | 5 |
| 4: | 0 | 0 | 0 | 0 | 0 | 0 |
| 5: | 0 | 0 | 0 | 2 | 0 | 2 |
| 6: | 0 | 0 | 0 | 0 | 0 | 0 |

| | bsyrct | rcyrct | bryrct | bsspan | rcspan | brspan |
|---|---|---|---|---|---|---|
| 1: | 11 | 0 | 11 | 11 | 0 | 11 |
| 2: | 6 | 0 | 6 | 6 | 0 | 6 |
| 3: | 10 | 0 | 10 | 10 | 0 | 10 |
| 4: | 17 | 27 | 44 | 30 | 27 | 57 |
| 5: | 10 | 0 | 10 | 10 | 0 | 10 |
| 6: | 28 | 11 | 39 | 30 | 11 | 41 |

Code 4 combines the output of Code 2 and 3, the detailed station inventory and the initial year-month summary. The code produces a year-month summary with station detail and similarly a detailed yearly summary. The year-month summary gains a more complete record count from the station detail; a month where records are missing or stations are not operating is merged to match the partial month of the data summary. The days in a month the station is not open are distinguished from the days the station is open but records are missing. The yearly summary further includes day counts for full months in the year where all records are missing or the station is closed.

## 5.5 Code 5: Missing Records by Year

---

Table 10. Code 5 Sample Output. Missing Records by Year. All sample weather elements, US and Canada.

| | loc | elem | year | zerrec | recs | zblank | zerobs | obs |
|---|---|---|---|---|---|---|---|---|
| 1: | US | AWND | 1982 | 0 | 243 | 0 | 0 | 243 |
| 2: | US | AWND | 1984 | 63 | 102531 | 0 | 63 | 102531 |
| 3: | US | AWND | 1985 | 111 | 101691 | 0 | 111 | 101691 |
| 4: | US | AWND | 1986 | 57 | 104826 | 0 | 57 | 104826 |
| 5: | US | AWND | 1987 | 46 | 116725 | 0 | 46 | 116725 |
| --- | | | | | | | | |
| 881: | CA | WSFG | 1969 | 0 | 364 | 0 | 0 | 364 |
| 882: | CA | WSFG | 1970 | 0 | 136 | 0 | 0 | 136 |
| 883: | CA | WSFG | 2015 | 70208 | 175879 | 0 | 70208 | 175879 |
| 884: | CA | WSFG | 2016 | 95246 | 235934 | 0 | 95246 | 235934 |
| 885: | CA | WSFG | 2017 | 74654 | 196613 | 0 | 74654 | 196613 |

| | stndays | stndysinyr | stnmos | clsdinm | clsdfulm | clsdall | misinm | misfulm | misall | stnct |
|---|---|---|---|---|---|---|---|---|---|---|
| 1: | 243 | 365 | 8 | 0 | 0 | 0 | 0 | 122 | 122 | 1 |
| 2: | 102968 | 105042 | 3376 | 16 | 640 | 656 | 421 | 1434 | 1855 | 287 |
| 3: | 102074 | 102930 | 3356 | 17 | 92 | 109 | 366 | 764 | 1130 | 282 |
| 4: | 105445 | 119355 | 3467 | 7 | 11421 | 11428 | 612 | 2489 | 3101 | 327 |
| 5: | 117619 | 118260 | 3867 | 1 | 579 | 580 | 893 | 62 | 955 | 324 |
| --- | | | | | | | | | | |
| 881: | 365 | 365 | 12 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 882: | 151 | 365 | 5 | 7 | 214 | 221 | 8 | 0 | 8 | 1 |
| 883: | 180861 | 247470 | 5919 | 536 | 64712 | 65248 | 4446 | 1897 | 6343 | 678 |
| 884: | 240462 | 247416 | 7884 | 379 | 3968 | 4347 | 4149 | 2986 | 7135 | 676 |
| 885: | 200453 | 239075 | 6597 | 779 | 36251 | 37030 | 3061 | 2371 | 5432 | 655 |

| | stndays | stndysinyr | stnmos | clsdinm | clsdfulm | clsdall | misinm | misfulm | misall |
|---|---|---|---|---|---|---|---|---|---|
| 1: | 243 | 365 | 8 | 0 | 0 | 0 | 0 | 122 | 122 |
| 2: | 102968 | 105042 | 3376 | 16 | 640 | 656 | 421 | 1434 | 1855 |
| 3: | 102074 | 102930 | 3356 | 17 | 92 | 109 | 366 | 764 | 1130 |
| 4: | 105445 | 119355 | 3467 | 7 | 11421 | 11428 | 612 | 2489 | 3101 |
| 5: | 117619 | 118260 | 3867 | 1 | 579 | 580 | 893 | 62 | 955 |
| --- | | | | | | | | | |
| 881: | 365 | 365 | 12 | 0 | 0 | 0 | 1 | 0 | 1 |
| 882: | 151 | 365 | 5 | 7 | 214 | 221 | 8 | 0 | 8 |
| 883: | 180861 | 247470 | 5919 | 536 | 64712 | 65248 | 4446 | 1897 | 6343 |
| 884: | 240462 | 247416 | 7884 | 379 | 3968 | 4347 | 4149 | 2986 | 7135 |
| 885: | 200453 | 239075 | 6597 | 779 | 36251 | 37030 | 3061 | 2371 | 5432 |

| | stnct | pctobsyr | pctmisyr | pctclsdyr | pctzblkyr | pctzeroyr | pctzerobs |
|---|---|---|---|---|---|---|---|
| 1: | 1 | 0.665753 | 0.334247 | 0 | 0 | 0 | 0 |
| 2: | 287 | 0.976095 | 0.01766 | 0.006245 | 0 | 0.0006 | 0.000614 |
| 3: | 282 | 0.987963 | 0.010978 | 0.001059 | 0 | 0.001078 | 0.001092 |
| 4: | 327 | 0.878271 | 0.025981 | 0.095748 | 0 | 0.000478 | 0.000544 |
| 5: | 324 | 0.98702 | 0.008075 | 0.004904 | 0 | 0.000389 | 0.000394 |
| --- | | | | | | | |
| 881: | 1 | 0.99726 | 0.00274 | 0 | 0 | 0 | 0 |
| 882: | 1 | 0.372603 | 0.021918 | 0.605479 | 0 | 0 | 0 |
| 883: | 678 | 0.710708 | 0.025631 | 0.26366 | 0 | 0.283703 | 0.399184 |
| 884: | 676 | 0.953592 | 0.028838 | 0.01757 | 0 | 0.384963 | 0.403698 |
| 885: | 655 | 0.822391 | 0.022721 | 0.154889 | 0 | 0.312262 | 0.3797 |

---

## 5.6 Code 6: Multiple Month Indices

---

Table 11. Code 6 Sample output. Multiple month indices (Bimonthly). Precipitation, US and Canada.

| | StationID | loc | St | lat | lon | elev | mindate | maxdate | elem |
|---|---|---|---|---|---|---|---|---|---|
| 1: | CA001010066 | CA | BC | 48.8667 | -123.283 | 4 | 19840701 | 19961129 | PRCP |
| 2: | CA001010066 | CA | BC | 48.8667 | -123.283 | 4 | 19840701 | 19961129 | PRCP |
| 3: | CA001010066 | CA | BC | 48.8667 | -123.283 | 4 | 19840701 | 19961129 | PRCP |
| 4: | CA001010066 | CA | BC | 48.8667 | -123.283 | 4 | 19840701 | 19961129 | PRCP |
| 5: | CA001010066 | CA | BC | 48.8667 | -123.283 | 4 | 19840701 | 19961129 | PRCP |
| --- | | | | | | | | | |
| 9184276: | USW00096406 | US | AK | 64.5014 | -154.13 | NA | 20140829 | 20171231 | PRCP |
| 9184277: | USW00096407 | US | AK | 66.562 | -159.004 | NA | 20150814 | 20171231 | PRCP |
| 9184278: | USW00096407 | US | AK | 66.562 | -159.004 | NA | 20150814 | 20171231 | PRCP |
| 9184279: | USW00096408 | US | AK | 63.4519 | -150.875 | NA | 20150820 | 20171214 | PRCP |
| 9184280: | USW00096408 | US | AK | 63.4519 | -150.875 | NA | 20150820 | 20171214 | PRCP |

| | year | ord | bimo | MEI | VAL2m | VAL2m_US | sumVALsqd_US | sumVALsqm_US |
|---|---|---|---|---|---|---|---|---|
| 1: | 1985 | 1 | JanFeb | -0.595 | 60 | 0.2362 | 0.042904086 | 5.58E-02 |
| 2: | 1986 | 1 | JanFeb | -0.195 | 2822 | 11.11 | 9.504433009 | 6.45E+01 |
| 3: | 1987 | 1 | JanFeb | 1.205 | 1332 | 5.2441 | 1.477462955 | 1.57E+01 |
| 4: | 1988 | 1 | JanFeb | 0.706 | 870 | 3.4252 | 1.73507347 | 8.60E+00 |
| 5: | 1989 | 1 | JanFeb | -1.262 | 1254 | 4.937 | 1.710893422 | 1.40E+01 |
| --- | | | | | | | | |
| 9184276: | 2016 | 12 | DecJan | 2.227 | 5 | 0.0197 | 0.000387501 | 3.88E-04 |
| 9184277: | 2015 | 12 | DecJan | 0.42 | 33 | 0.1299 | 0.008137516 | 1.69E-02 |
| 9184278: | 2016 | 12 | DecJan | 2.227 | 111 | 0.437 | 0.034425569 | 9.56E-02 |
| 9184279: | 2015 | 12 | DecJan | 0.42 | 0 | 0 | 0 | 0.00E+00 |
| 9184280: | 2016 | 12 | DecJan | 2.227 | 470 | 1.8504 | 0.351695703 | 1.72E+00 |

```
         zerrec    recs  zblank  zerobs    obs     daysum    misinm  clsdinm   monthct
 1:          23      25       0      23     25         31         6        0         1
 2:          27      58       0      27     58         59         1        0         2
 3:          25      55       0      25     55         59         4        0         2
 4:          36      51       0      36     51         60         9        0         2
 5:          24      49       0      24     49         59        10        0         2
 ---
 9184276:     0       1       0       0      1         31        30        0         1
 9184277:    19      23       0      19     23         62        39        0         2
 9184278:    16      24       0      16     24         62        38        0         2
 9184279:     1       1       0       1      1         31        30        0         1
 9184280:    39      60       0      39     60         62         2        0         2

-----------------------------------------------------------------------------------------
ord       sorting order for multiple month time period based on first month of period
bimo      bimonthly time period for MEI (JanFeb, FebMar, …, NovDec)
trimo     (not shown) trimonthly time period for ONI (JFM, FMA, …, DJF)
MEI       bimonthly Multi-Variate ENSO Index
ONI       (not shown) trimonthly Ocean Niño Index
misinm    sum of missing records for combined months of bimo or trimo period
clsdinm   sum of days station closed for combined months of bimo or trimo period
monthct   count of months with records for bimo or trimo period
```

Some ENSO indices are based on multiple months of data. Bimonthly sums are merged with the Multivariate ENSO Index (MEI) and trimonthly sums are merged with the Oceanic Niño Index (ONI). A count of months is made to indicate at least one record was present in each month. The detail of observations by month can be viewed in the year-month summary. Stations may be selected having observations near to 60 (bimonthly) or 90 (trimonthly) or a minimum monthly observation can be established utilizing the year-month summary for station selection.

### 5.7 Code 7: Weekly Niño Indices

```
-----------------------------------------------------------------------------------------
Table 12.  Code 7 Sample output.  Weekly Niño indices.  Precipitation, US and Canada.

        StationID     loc  elem  St    lat       lon      elev  weekno  yrgrp  ctrweek
 1:     CA001010066   CA   PRCP  BC    48.8667  -123.283     4  1566    1991   1/2/1991
 2:     CA001010960   CA   PRCP  BC    48.6     -123.467    38  1566    1991   1/2/1991
 3:     CA001011467   CA   PRCP  BC    48.5833  -123.417    53  1566    1991   1/2/1991
 4:     CA0010114F6   CA   PRCP  BC    48.5667  -123.4      38  1566    1991   1/2/1991
 5:     CA001011743   CA   PRCP  BC    48.6833  -123.6      99  1566    1991   1/2/1991
 ---
 544315: USW00094911  US   PRCP  SD    42.8783   -97.3633   NA  1617    1991   12/25/1991
 544316: USW00094918  US   PRCP  NE    41.3536   -96.0233   NA  1617    1991   12/25/1991
 544317: USW00094931  US   PRCP  MN    47.3864   -92.8389   NA  1617    1991   12/25/1991
 544318: USW00094957  US   PRCP  NE    40.0803   -95.5919   NA  1617    1991   12/25/1991
 544319: USW00094967  US   PRCP  MN    46.9006   -95.0678   NA  1617    1991   12/25/1991

         VALw    VALw_US     sumVALsqd_US  VALsqw_US   zerrec  recs  zblank  zerobs  obs  misclsd
 1:        0     0           0             0              6      6      0       6     6      8
 2:       85     0.3346457   0.04738359    0.07168764     4      7      0       4     7      7
 3:      140     0.5511811   0.13751628    0.2015004      4      7      0       4     7      7
 4:       96     0.3779528   0.06782814    0.08642817     4      7      0       4     7      7
 5:       54     0.2125984   0.02287805    0.02287805     4      6      0       4     6      8
 ---
```

```
 544315:    0      0           0           0           7     7     7     0    0     7
 544316:  145      0.5708661   0.31268213  0.32588815  5     7     0     5    7     0
 544317:    0      0           0           0           7     7     0     7    7     0
 544318:  117      0.4606299   0.2015779   0.21217992  5     7     0     5    7     0
 544319:    0      0           0           0           7     7     0     7    7     0

         Nino12Ind  Nino12Anom  Nino3Ind   Nino3Anom  Nino34Ind  Nino34Anom  Nino4Ind Nino4Anom
 1:         23.2       0.5         25.3       0.1        26.9       0.4         28.9     0.5
 2:         23.2       0.5         25.3       0.1        26.9       0.4         28.9     0.5
 3:         23.2       0.5         25.3       0.1        26.9       0.4         28.9     0.5
 4:         23.2       0.5         25.3       0.1        26.9       0.4         28.9     0.5
 5:         23.2       0.5         25.3       0.1        26.9       0.4         28.9     0.5
 ---
 544315:    23.6       0.3         26.7       1.4        28.5       1.9         29.6     1.2
 544316:    23.6       0.3         26.7       1.4        28.5       1.9         29.6     1.2
 544317:    23.6       0.3         26.7       1.4        28.5       1.9         29.6     1.2
 544318:    23.6       0.3         26.7       1.4        28.5       1.9         29.6     1.2
 544319:    23.6       0.3         26.7       1.4        28.5       1.9         29.6     1.2
--------------------------------------------------------------------------------------------
VALw             weekly sum of daily VAL
VALw_US          weekly sum of VAL_US, VAL converted to U.S. unit of measure.
sumVALsqd_US     weekly sum of squared daily values in U.S. units, to compute daily variances
```

Niño indices are available monthly at least since 1950, and weekly since 1990. The weekly indices give more specific information about the ENSO phase that might be relevant to the timing of losses. The code produces weekly summaries by element according to the weekly groupings of the Niño indices, which begin on 12/31/1989 and are always seven days in length. The result of these divisions will be the same beginning on 1/1/1961. As these weeks will shift through years, they will not provide comparison periods year by year. An alternative numbering scheme is provided at the end of the code labels an eight day week at the end of each year and an eight day week with every leap day. The eighth days weeks can be excluded from assignment if desired. This alternative provides a comparative basis among years but does not match the weekly Niño indices time periods so the index values might be interpolated. The code counts the number of observations by week so that the level of completeness can be determined and used in station selection.

```
--------------------------------------------------------------------------------------------
Table 13.  Weekly Niño numbering scheme (Table 13-A, left) with seven days per week, and
alternative scheme for comparisons among years (Table 13-B, right) with eight day weeks at the
end of each year and with each leap day.

      weekno   date        ctrweek      yrgrp       ||          weekno    mdchar    ctrweek
  1      1     19610101    1961-01-04   1961        ||    1       1        101      1961-01-04
  2      1     19610102    1961-01-04   1961        ||    2       1        102      1961-01-04
  3      1     19610103    1961-01-04   1961        ||    3       1        103      1961-01-04
  4      1     19610104    1961-01-04   1961        ||    4       1        104      1961-01-04
  5      1     19610105    1961-01-04   1961        ||    5       1        105      1961-01-04
  6      1     19610106    1961-01-04   1961        ||    6       1        106      1961-01-04
  7      1     19610107    1961-01-04   1961        ||    7       1        107      1961-01-04
  8      2     19610108    1961-01-11   1961        ||    8       2        108      1961-01-11
  9      2     19610109    1961-01-11   1961        ||    9       2        109      1961-01-11
  10     2     19610110    1961-01-11   1961        ||    10      2        110      1961-01-11
  11     2     19610111    1961-01-11   1961        ||    11      2        111      1961-01-11
  12     2     19610112    1961-01-11   1961        ||    12      2        112      1961-01-11
--------------------------------------------------------------------------------------------
```

```
weekno     (A) each week is given its own number; (B) weeks 1 – 52 each year
date       (A) always 7 days in week, field to merge with element or loss   records
ctrweek    weeks are identified by central day of the week.  (A) matches Niño indices.
yrgrp      indicates output *.csv file particularly for weeks overlapping two years
mdchar     (B) month-day will correspond to the same week number in every year.
```

## 5.8 Code 8 State Summaries
## Visual Analysis with Choropleth Maps

Table 14. Code 8 sample of summary by state, with anomalies to be plotted on choropleth maps. Precipitation, US by state.

| | loc | St | elem | year | VALy | VALy_US | obs | stnct | VALsqSt_US |
|---|---|---|---|---|---|---|---|---|---|
| 1: | US | AL | PRCP | 1960 | 487419 | 1918.9724 | 14162 | 39 | 3682455.2 |
| 2: | US | AZ | PRCP | 1960 | 159360 | 627.4016 | 21350 | 60 | 393632.7 |
| 3: | US | AR | PRCP | 1960 | 796665 | 3136.4764 | 24788 | 70 | 9837484.1 |
| 4: | US | CA | PRCP | 1960 | 1083230 | 4264.685 | 75153 | 208 | 18187538.5 |
| 5: | US | CO | PRCP | 1960 | 246999 | 972.437 | 26517 | 74 | 945633.7 |
| 6: | US | CT | PRCP | 2016 | 74192 | 292.0945 | 2921 | 8 | 85319.19 |

| | bsstnobs | bsstnyrs | bsmean | bsstdev | rm.na | anom |
|---|---|---|---|---|---|---|
| 1: | 396676 | 1200 | 2239.84 | 319.3026 | TRUE | -1.0049014 |
| 2: | 544398 | 1800 | 1050.5949 | 1052.0216 | TRUE | -0.4022668 |
| 3: | 691713 | 2130 | 3467.5654 | 556.574 | TRUE | -0.5948697 |
| 4: | 1874258 | 6270 | 4607.6652 | 1421.5972 | TRUE | -0.241264 |
| 5: | 722090 | 2280 | 1122.9764 | 165.4945 | TRUE | -0.9096334 |
| 6: | 79998 | 240 | 366.5045 | 61.08135 | TRUE | -1.2182111 |

| | Name | STDEVgrp | colreg |
|---|---|---|---|
| 1: | alabama | neg1.5 | wheat2 |
| 2: | arizona | neg0.5 | lightyellow1 |
| 3: | arkansas | neg0.5 | lightyellow1 |
| 4: | california | pos0.50 | lightcyan1 |
| 5: | colorado | neg0.5 | lightyellow1 |
| 6: | connecticut | neg1.5 | wheat2 |

Code 8 summarizes by state, greatly reducing the data set size, and from the state data creates choropleth maps – maps that color code a region according to a selected data field. Two different packages are used to create choropleth maps, maps that color code a selected data field by region. The packages are 'maps' and 'ggplot2.' In this case the regions are states, and the example data field is the anomalous rainfall in the year mapped as compared against the base climate period. Choropleths of anomalies centered at zero are not as immediately plotted in R because the choropleth packages are primarily designed for positive values such as census data.

Package 'maps' does not plot Alaska or Hawaii with the mainland United States, but the code is simple and straightforward to use for creating a choropleth, and individual states can still be plotted separately. The example choropleth will be created 'from scratch,' meaning a column of colors will be defined in the data set corresponding to the anomaly for each state. In this case, the positive anomalies will be assigned deepening shades of blue to indicate more rainfall at a glance, while the negative anomalies will be assigned deepening shades of tan and brown to indicate dryness. Points are easily plotted on the choropleth to show the location of each station selected for the data underlying the choropleth.

Package 'ggplot2' has an advantage over 'maps' in its compatibility with the package 'fiftystater' which plots insets for Alaska and Hawaii. This package also allows a midpoint to be defined at zero, and will either assign automatic base colors or else will assign grades of colors based on selections made for the low, midpoint, and high values. Selecting a midpoint color with contrast to the low and high value colors will produce a choropleth that is easily interpreted.

To avoid plot errors and formatting glitches, be sure to expand the plot region large enough for map to fit, and for the legend to fit to the side of the map.

The code uses package 'dplyr' to produce statistics for the base climate period, but also provides the formulas for calculating anomalies directly. Note that if the package 'plyr' is loaded in R, then 'dplyr' will not complete the calculations unless 'plyr' is detached.

---

Table 15. Code 8 sample calculation of base year statistics, mean and standard deviation. Precipitation, US by state.

| | loc | St | elem | bsstnobs | bsstnyrs | bsmean | bsstdev | rm.na |
|---|---|---|---|---|---|---|---|---|
| | <chr> | <chr> | <chr> | <int> | <dbl> | <dbl> | <dbl> | <lgl> |
| 1 | US | AK | PRCP | 348076 | 990 | 1354.76 | 118.8982 | TRUE |
| 2 | US | AL | PRCP | 396676 | 1200 | 2239.84 | 319.3026 | TRUE |
| 3 | US | AR | PRCP | 691713 | 2130 | 3467.565 | 556.574 | TRUE |
| 4 | US | AZ | PRCP | 544398 | 1800 | 1050.595 | 1052.0216 | TRUE |
| 5 | US | CA | PRCP | 1874258 | 6270 | 4607.665 | 1421.5972 | TRUE |
| 6 | US | CO | PRCP | 722090 | 2280 | 1122.976 | 165.4945 | TRUE |

---

## 5.9 Code 9 Combine Monthly Indices

---

Table 16. Code 9 Sample Output. Monthly ENSO indices combined into a single file 'IndexMonthly.csv' with all indices converted to 'long' format.

| | year | month | Nino12 | Anom12 | Nino3 | Anom3 | Nino4 | Anom4 | Nino34 | Anom34 | SOI | EQSOI | BEST | TNI |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | <int> | <int> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> |
| 1 | 1951 | 1 | 24.11 | -0.44 | 24.79 | -0.87 | 27.21 | -1.02 | 25.24 | -1.31 | 2.5 | 0.1 | -1.13 | 1.315 |
| 2 | 1951 | 2 | 25.19 | -0.83 | 25.65 | -0.76 | 27.09 | -1.01 | 25.71 | -1.03 | -1.5 | -1.4 | 0.64 | 0.168 |
| 3 | 1951 | 3 | 25.74 | -0.68 | 26.87 | -0.28 | 27.74 | -0.47 | 26.90 | -0.33 | 0.5 | -0.2 | 0.18 | -0.027 |
| 4 | 1951 | 4 | 25.29 | -0.18 | 27.37 | -0.11 | 28.21 | -0.24 | 27.58 | -0.13 | 1.1 | 0.2 | 0.00 | -0.655 |
| 5 | 1951 | 5 | 24.59 | 0.33 | 27.07 | -0.09 | 29.18 | 0.43 | 27.92 | 0.11 | -0.9 | -0.2 | -0.01 | 0.316 |

---

This interim code combines various ENSO index files into one, for convenience, utilizing a common "long" format. The Niño indices are promulgated in the "long" format while other indices are available formatted "wide." For ENSO indices, the "long" format includes two separate columns for year and month, and one column per index; while the "wide" format includes separate columns for each month of the year. It is most practical to utilize the "long" format for all indices, in order to merge data by year and month, and to label each index as a column header.

## 5.10 Code 10 Plot ENSO Index Time Series

```
---------------------------------------------------------------------------------------------
Table 17. Sample output of Multivariate ENSO Index (MEI) for plotting time series.

        bimo        Year      MEI       posM      month
 273    JanFeb      1950    -1.163      FALSE       1
 205    FebMar      1950    -1.312      FALSE       2
 477    MarApr      1950    -1.098      FALSE       3
 1      AprMay      1950    -1.445      FALSE       4
 545    MayJun      1950    -1.376      FALSE       5

---------------------------------------------------------------------------------------------
```

This code produces time series bar plots of indices from MEI, ONI, and the monthly indices previously combined into one dataset by Code 9. The column "positive" is added to discern index values above and below the x-axis, corresponding to positive and negative ENSO phases color coded in the bar graph. The plots are formatted for anomalies and could be modified to present SST measures

For the monthly ENSO indices, a vector "IndexName" is defined by the column headings for the index values. The ENSO index to be plotted is selected by its number position in the vector. The code replaces the column heading with "PlotIndex" which locates the data to plot. After plotting the column heading is returned to the original ENSO index title. If errors are encountered, reset the column headers to the original headers.

```
------------------------------------------------------------------------------------------
Table 18.  Column headings for data set 'indices' given by names(indices), before
selecting the index (top) and after, where SelIndex <- 4 (bottom).

  [1] "year"   "month"   "Nino12"  "Anom12"  "Nino3"   "Anom3"     "Nino4"
  [8] "Anom4"  "Nino34"  "Anom34"  "SOI"     "EQSOI"   "BEST"      "TNI"

  [1] "year"   "month"   "Nino12"  "Anom12"  "Nino3"   "PlotIndex" "Nino4"
  [8] "Anom4"  "Nino34"  "Anom34"  "SOI"     "EQSOI"   "BEST"      "TNI"
------------------------------------------------------------------------------------------
```

## 5.11 Code 11 Plot Element vs. Index by State

```
---------------------------------------------------------------------------------------------
Table 19.  Sample output of monthly data summarized by state/territory with corresponding
monthly ENSO indices.  Precipitation, US and Canada.

      St       loc      elem       year    month      pre82       VALm      VALm_US
 1:   BC       CA       PRCP       1960      1          Y         40321     158.74409
 2:   BC       CA       PRCP       1960      2          Y         32910     129.56693
 3:   BC       CA       PRCP       1960      3          Y         30385     119.62598
 4:   BC       CA       PRCP       1960      4          Y         27480     108.18898
 5:   BC       CA       PRCP       1960      5          Y         26997     106.2874
 6:   BC       CA       PRCP       1960      6          Y         11381     44.80709
```

```
      sumVALsqd_US     VALsqm_US     zerrec    recs     zblank   zerobs     obs    daysinmo
 1:     192.12355       1613.4087      429      863        0      429       863      868
 2:     142.06836       1054.816       467      806        0      467       806      812
 3:     86.56722        882.4508       452      849        0      452       849      868
 4:     79.6731         826.5286       488      856        0      488       856      870
 5:     61.02212        601.5856       404      875        0      404       875      899
 6:     15.57885        109.3986       570      838        0      570       838      840

      misinm    clsdinm    stns     avgVALm_US       Nino12      Anom12     Nino3      Anom3
 1:      5         0        28     0.18394449        24.23       -0.31      25.31     -0.35
 2:      6         0        28     0.16075301        25.68       -0.34      25.93     -0.47
 3:     19         0        28     0.14090222        26.24       -0.18      26.87     -0.29
 4:      8         6        29     0.12638899        24.43       -1.04      27.15     -0.33
 5:     24         0        29     0.12147132        23.33       -0.94      26.71     -0.45
 6:      2         0        28     0.05346908        21.71       -1.3       25.86     -0.64

      Nino4       Anom4      Nino34      Anom34      SOI       EQSOI       BEST       TNI
 1:    27.62       -0.62      26.27       -0.29      -1.5       -1.1        1.51     -0.945
 2:    27.44       -0.65      26.29       -0.45      -1.2       -0.7        0.74     -0.668
 3:    27.75       -0.45      26.98       -0.25       0.4        0.4       -0.07     -1.399
 4:    28.01       -0.44      27.49       -0.22      -0.2       -0.1        0.7      -1.911
 5:    28.42       -0.33      27.68       -0.13       0.4        0.7       -0.39     -0.373
 6:    28.33       -0.46      27.24       -0.35       2.9        0.3       -0.75     -1.149
-----------------------------------------------------------------------------------------------
```

Code 11 summarizes by state, greatly reducing the data size. This code is intended to be highly customizable. The example given is a simplified illustration. Yearly data is used to select stations with records in all 57 base and recent years, which adds consistency to the location of observations across time so that comparisons of yearly results are meaningful. The detailed station inventories could be used to make this type of selection, but reading yearly data has the advantage including counts of observations by which to further refine selections. Alternatively, the year-month summary could be used to set a minimum level of completeness for selected months based on monthly observation counts. Many other criteria can be introduced. As the weather element data will be summarized from a station level to a state level, it is important to consider the stations represented by the selections. Strict selections may result in overly sparse records by state or sparseness in relevant regions.

The data to read in will be monthly, bimonthly or trimonthly, depending on the index selection. The MEI is bimonthly and ONI is trimonthly. The vector "IndexName" is defined by the column headings for index values, the same as in Code 10. A loop is coded so that all of the indices can be plotted at once; or the loop can be commented out. If errors are encountered, remember to reset the column headers to the original headers.

The code identifies years prior to 1982 versus years from 1982 on, and base climate period years versus subsequent years. The data is summed preserving the 1982 split but could be modified to retain the base period, or another split specified by a code modification. Outliers labeled to identify the year of each point, can be revised at the left and right limits, according to the overall spread of the plot. Some graph labels are also automated and may need to be refined.

For the example, no adjustment is made for the change in proportion of blank records assumed zero ('zblank') which has been drastic especially since 1982. The plot is color-coded to show points before and after 1982. Erroneous zero entries will be overstated prior to 1982, so that adjustments for unidentified blanks assumed zero would shift points upward. The adjustment should affect the final relationship displayed between the weather element and the ENSO index. The effort to improve completion of records since 1982 will also cause zero records to increase since 1982, over which time zero entries have been somewhat understated.

### 5.12 Code 12 Map of ENSO Index Regions

This code creates a map of the ENSO Index regions. Details on mapping are covered in Part I.

### 5.13 Code 13 Costliest Storms

The data on costliest storms was copied from Wikipedia in January 2018. The table below can be copied into excel and saved as a *.csv file 'CostlyStorms.csv' in the base directory as input to this code. The Wikipedia data is updated regularly, and if the input file is updated then the formatting of the bar graph will require updates to the code. Some storm dates are provide which can be used for comparison against ENSO indices.

**Table 20. List of Costliest Atlantic Hurricanes.** Storms exceeding U.S. $1 Billion, in descending order. Storms that broke the historical record for damages, at the time of the storm's dissipation, are highlighted, showing that the costliest storms have move up the list in large strides that may appear uncharacteristic of inflation or randomness by damages on an unadjusted actual cost level. This table is intended for input to R. [ Source : Wikipedia ]

| Storm Name | Peak Classification Hurricane Category (0 = Tropical Storm) | Unadjusted Damages in U.S. $Billions | Year | (> $1B) Storm # of Year | Begin Date | End Date |
|---|---|---|---|---|---|---|
| **Katrina** | **5** | **125** | **2005** | **3** | **823** | **831** |
| **Harvey** | **4** | **125** | **2017** | **1** | **817** | **903** |
| Maria | 5 | 92 | 2017 | 3 | 916 | 1003 |
| Sandy | 3 | 68.7 | 2012 | 2 | 1022 | 1102 |
| Irma | 5 | 64.2 | 2017 | 2 | 830 | 916 |
| Ike | 4 | 38 | 2008 | 3 | 901 | 915 |
| Wilma | 5 | 27.4 | 2005 | 6 | 1016 | 1027 |
| **Andrew** | **5** | **27.3** | **1992** | **1** | **816** | **828** |
| Ivan | 5 | 26.1 | 2004 | 3 | 902 | 924 |
| Rita | 5 | 18.5 | 2005 | 4 | 918 | 926 |
| Charley | 4 | 16.9 | 2004 | 1 | 809 | 815 |
| Matthew | 5 | 15.1 | 2016 | 1 | 928 | 1010 |
| Irene | 3 | 14.2 | 2011 | 1 | 821 | 830 |
| Frances | 4 | 9.8 | 2004 | 2 | 824 | 910 |
| **Hugo** | **5** | **9.47** | **1989** | **1** | **910** | **925** |
| Georges | 4 | 9.37 | 1998 | 2 | 915 | 1001 |
| Allison | 0 | 8.5 | 2001 | 1 | 604 | 620 |
| Gustav | 4 | 8.31 | 2008 | 2 | 825 | 907 |
| Jeanne | 3 | 7.94 | 2004 | 4 | 913 | 929 |
| Floyd | 4 | 6.5 | 1999 | 1 | 907 | 919 |
| Mitch | 5 | 6.08 | 1998 | 3 | 1022 | 1109 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Isabel | 5 | 5.5 | 2003 | 1 | 906 | 920 |
| Fran | 3 | 5 | 1996 | 1 | 823 | 910 |
| Opal | 4 | 4.7 | 1995 | 3 | 927 | 1006 |
| Stan | 1 | 3.96 | 2005 | 5 | 1001 | 1005 |
| Karl | 3 | 3.9 | 2010 | 2 | 914 | 918 |
| Dennis | 4 | 3.71 | 2005 | 1 | 704 | 718 |
| **Alicia** | **3** | **3** | **1983** | **1** | **815** | **821** |
| Gilbert | 5 | 2.98 | 1988 | 1 | 908 | 929 |
| Luis | 4 | 2.97 | 1995 | 1 | 828 | 912 |
| Lee | 0 | 2.8 | 2011 | 2 | 902 | 907 |
| Isaac | 1 | 2.8 | 2012 | 1 | 821 | 903 |
| Michelle | 4 | 2.35 | 2001 | 2 | 1029 | 1106 |
| **Agnes** | **1** | **2.1** | **1972** | **1** | **614** | **623** |
| Marilyn | 3 | 2.1 | 1995 | 2 | 912 | 930 |
| Dean | 5 | 1.95 | 2007 | 1 | | |
| Alex | 2 | 1.89 | 2010 | 1 | | |
| Joan | 4 | 1.87 | 1988 | 2 | | |
| Fifi | 2 | 1.8 | 1974 | 1 | | |
| Frederic | 4 | 1.71 | 1979 | 2 | | |
| Dolly | 2 | 1.6 | 2008 | 1 | | |
| Allen | 5 | 1.57 | 1980 | 1 | | |
| David | 5 | 1.54 | 1979 | 1 | | |
| Bob | 3 | 1.51 | 1991 | 1 | | |
| Juan | 1 | 1.5 | 1985 | 2 | | |
| Roxanne | 3 | 1.5 | 1995 | 4 | | |
| Ingrid | 1 | 1.5 | 2013 | 1 | | |
| **Betsy** | **4** | **1.43** | **1965** | **1** | | |
| Camille | 5 | 1.42 | 1969 | 1 | | |
| Elena | 3 | 1.3 | 1985 | 1 | | |
| Isidore | 3 | 1.28 | 2002 | 1 | | |
| Lili | 4 | 1.16 | 2002 | 2 | | |
| Alberto | 0 | 1.03 | 1994 | 1 | | |
| Emily | 5 | 1.01 | 2005 | 2 | | |
| Beulah | 5 | 1 | 1967 | 1 | | |
| Bonnie | 3 | 1 | 1998 | 1 | | |

# Code

```
# ==================================================== #
# ================= CODE CONTENTS =================== #
# ==================================================== #

# 5.0                 Set up in R
# 5.1   Code 1        Weather Daily - Loop to Unzip Year by Year
# 5.2   Code 2        Initial Detailed Station Inventories
# 5.3   Code 3        Initial Year-Month Summaries
# 5.4   Code 4        Complete Yearly and Year-Month Summaries
#                            Merged with Station Locations and Inventories
# 5.5   Code 5        Missing Records by Year
# 5.6   Code 6        Multiple Month Indices (MEI and ONI)
# 5.7   Code 7        Weekly Nino Indices
# 5.8   Code 8        State Summaries / Plot Selected Stations
#                         Visual Analysis with Choropleth Maps
#                            5.8.1 Package 'maps' - 48 mainland states
#                            5.8.2 Packages 'ggplot2' and 'fiftystater' - AK & HI insets
# 5.9   Code 9        Combine Monthly Indices
# 5.10  Code 10       Plot Index Time Series
# 5.11  Code 11       Plot Element vs. Index by State
# 5.12  Code 12       Map of ENSO Index Regions
# 5.13  Code 13       Costliest Storms


# Station/Element level:  1 (daily), 2, 3 (yr-mo), 4 (yr-mo & yr), 6 (2 mo & 3 mo), 7 (weekly)
# Country/Element level:  5
# State/Element level:    8, 10
#
# 1. Loop to unzip daily meteorological data year by year.  Long run time.
# 2. Open daily files to detail station inventories, dates open/closed, missing records.
# 3. Open daily files to summarize by year-month.
# 4. Merge station detail into year-month summary; summarize by year
# 5. Yearly summary of total and average counts of missing records, blanks assumed zero,
#      observations, etc.
# 6. Summarize monthly data into two- and three-month periods for comparison to MEI and ONI.
# 7. Open daily files to summarize by seven day periods matching weekly Nino indices.
# 8. Customizable station selections, summarize by state, plot stations and choropleth
# 9. Convert wide formats to long; combine monthly ENSO indices into one .csv file for
#      convenience.
#
# ENSO Indices - bimonthly (MEI), trimonthly (ONI), monthly (Nino, SOI, EQSOI, TNI, BEST),
# and weekly (Nino)

# Daily GHCNDex data - download files by year into a folder specified as the working directory
# ftp://ftp.ncdc.noaa.gov/pub/data/ghcn/daily/
# ftp://ftp.ncdc.noaa.gov/pub/data/ghcn/daily/by_year/

# ==================================================== #
# ================= SET UP IN R ===================== #
# ==================================================== #


########################################
#             BEGIN SETUP              #
########################################

# Set Default Working Directory (Optional)
```

```
setwd("C:/…/Weather")
getwd()

# Remove list to free memory
rm(list=ls())
ls()

# Set directories for downloaded zipped files, base input files, and written output
dirzip <- "C:/…/WeatherZip"
dirbase <- "C:/…/WeatherBase"
diroutput <- "C:/…/WeatherData"

# Load packages data.table, tidyverse, lubridate
library(data.table)   # data.table functions run faster than base R code.
# rbindlist() combines years of weather dataframes in list;
# setnames() updates column headers
library(tidyverse)    # a set of packages for organizing data; package 'readr' to unzip.
library(lubridate)    # days_in_month() gives expected number of records


#................ FUNCTION REPEAT ROWS ..............
rep.row <- function(x,n){
  matrix(rep(x,each=n),nrow=n)
}
#....................................................


Elements <- c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN', 'WIND')

#----------------- FIVE CORE ELEMENTS ---------------------------------------
#
# SelElem     Element           Unit of Measure        Converted (US)
#
# PRCP Precipitation        tenths of mm          inches
# SNOW Snowfall                 mm                    inches
# SNWD Snow depth             mm                    inches
# TMAX Maximum temperature    tenths of degrees C    degrees Fahrenheit
# TMIN Minimum temperature    tenths of degrees C    degrees Fahrenheit
#
#---------------------------------------------------------------------------
#
# WIND    elements are coded to include:
#
# AWND    Average daily wind speed         (tenths of meters per second)
# WSF1    Fastest 1-minute wind speed      (tenths of meters per second)
# WSF2    Fastest 2-minute wind speed      (tenths of meters per second)
# WSF5    Fastest 5-second wind speed      (tenths of meters per second)
# WSFG    Peak gust wind speed             (tenths of meters per second)
# WSFI    Highest instantaneous wind speed (tenths of meters per second)
# WSFM    Fastest mile wind speed          (tenths of meters per second)
#
#----------------------------------------------------------------------------


########################################
#            END SETUP                 #
########################################

#
==============================================================================================
#
```

```
# ===== CODE 1 ===== LOOP TO UNZIP DAILY METEOROLOGICAL FILES YEAR BY YEAR
==================== #
#
==============================================================================================
#

# Elements                    # View list of elements
# Elements <- Elements[1:2]   # Select subset of elements (option)

# Set directory to file location of downloaded zipped files
setwd(dirzip)
# Put files in directory that will be unzipped and read
gzfiles <- dir(pattern = "*.csv.gz") # creates the list of all the csv files in the directory
gzfiles   # view files selected
# gzfiles <- gzfiles[1:3]  # Select subset of files from the list (option)

dataset <- list() # creates a list that will hold the meteorological data files

########################################
#          BEGIN OUTER LOOP           #
########################################

# OUTER LOOP : Selected years of zipped .GZ daily weather station data files

for (k in 1:length(gzfiles)){
  setwd(dirzip)
  dataset[[k]] <- read_csv(gzfiles[k], col_names = FALSE)
  # Add column names
  colnames(dataset[[k]]) <- c("StationID","date", "elem", "VAL", "MFlag",
                              "QFlag", "SFlag", "Time" )

  # Create data table to save processing time - Subset elements from this table
  dtbl <- as.data.table(dataset[[k]])

  # reduce the large data frame in the list to save memory in the loop
  dataset[[k]] <- 0

  # Create location field (country etc.) to be used to subset data
  dtbl[, loc := substring(StationID, 1, 2)]
  class(dtbl$elem)

  ########################################
  #          BEGIN INNER LOOP           #
  ########################################

  # Inner Loop : all Elements for the unzipped year

  for (L in 1: length(Elements)){
    # 'WIND' will subset several wind elements; otherwise use SelElem
    ifelse(Elements[L] != 'WIND',  SelElem <- Elements[L],
           SelElem <- c("AWND", "WSF1", "WSF2", "WSF5", "WSFG", "WSFI", "WSFM"))

    # Subset US and Canadian data for selected element, so datasets are small enough to write
    subdat <- dtbl[elem %in% SelElem & loc %in% c('US', 'CA')]
    subdat[is.na(subdat)]<- "-"
    # Replace line above to include US Territories
    # subdat <- dtbl[elem %in% SelElem & loc %in% c('US', 'CA', 'AQ', 'CQ', 'GQ',
    #'JQ', 'LQ', 'RQ', 'VQ', 'WQ')]

# Add year and month fields
subdat[, year := as.integer(substring(date, 1, 4))]
```

```
subdat[, month := as.integer(substring(date, 5, 6))]
subdat[, monthday := as.character(substring(date, 5, 8))]
# N.B. monthday 0122 appears in csv as 122. 1202 appears in csv as 1202.

# Convert unit of measure specific to selected element.
if(SelElem == 'PRCP'){
  subdat[,VAL_US := (VAL/254)]
}
if(SelElem == 'SNOW' | SelElem == 'SNWD'){
  subdat[,VAL_US := (VAL/25.4)]
}
if(SelElem == 'TMAX' | SelElem == 'TMIN'){
  subdat[,VAL_US := (VAL*0.18) +32]
}
if(Elements[L] == 'WIND'){
  subdat[,VAL_US := (VAL/10)*2.23694]
}

# Sort the files by location and StationID
subdat <- subdat[order(subdat$elem, subdat$StationID), ]

# Create file name according to year of data
yrchar = as.character(subdat[2, 10])

# Name file where daily data will be written out to
filenmday = paste0("USCANday", Elements[L], yrchar, ".csv")
# Write DAILY subsets of data to csv files
setwd(diroutput)
write_csv(subdat, filenmday, col_names=TRUE)

# Remove datasets and unused values to save memory
rm(filenmday, yrchar, subdat)

gc()  # call for garbage can saves memory
  }

  ########################################
  #      END INNER LOOP (Elements)       #
  ########################################

  rm(dtbl)

}

########################################
#       END OUTER LOOP (Years)        #
########################################

rm(dataset, gzfiles, k, L, SelElem)

# END PROGRAM CODE

#

# ========================================================================================= #
# ===== CODE 2 =============== INITIAL STATION INVENTORY ================================= #
# ========================================================================================= #

# Go back to repeat SETUP at top if R has been closed.

# Select subset of elements (option)
```

```
Elements <- c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN', 'WIND')
# Elements <- Elements[1:2]

# Select base years (eg. 1961 - 1990) for climatology, typically 30 past years
BegBsYr <- 1961
EndBsYr <- 1990

setwd(dirbase)

stnlist <- read_csv('ghcnd-stations.csv', col_names = FALSE)
colnames(stnlist) <- c("StationID","lat", "lon", "elev", "St", "Name", "GSNFlag", "zip" )
stnlist$loc <- as.character(substring(stnlist$StationID, 1,2))
stnlist <- as.data.table(stnlist)
stnsub <- stnlist[, c('StationID', 'loc', 'St', 'lat', 'lon', 'elev')]  #(*Change columns*)
USCANstn <- subset(stnsub, stnsub$loc %in% c('US', 'CA'))
head(USCANstn)


stninv <- read_csv('ghcnd-inventory.csv', col_names = FALSE)
colnames(stninv) <- c("StationID","lat", "lon", "elem", "Open", "Close")
stninv$loc <- as.character(substring(stninv$StationID, 1,2))
stninv <-as.data.table(stninv)
#(*Change columns*)
stninv0 <- stninv[, c('StationID', 'loc','lat', 'lon', 'elem', 'Open', 'Close')]
USCANinv0 <- stninv0[loc %in% c("US", "CA")]
rm(stnlist, stnsub, stninv, stninv0)

# Set working directory to access output of daily csv files
setwd(diroutput)
########################################
#         BEGIN OUTER LOOP             #
########################################
for (w in 1:length(Elements)){
  SelElem <- Elements[w]
  # Select station inventory by element, merge with station list
  USCANinv <- USCANinv0[elem == SelElem]
  stnloc <- as.data.table(full_join(USCANstn, USCANinv[, c('StationID', 'Open', 'Close')],
                                   by = 'StationID'))

  # creates the list of all the csv files in the directory
  csvfiles <- dir(pattern = paste0("USCANday",  SelElem, "*"))
  csvfiles

  # Define list for loop
  daily <- list()

  ########################################
  #         BEGIN INNER LOOP             #
  ########################################

  ##### LOOP: for selected element, loop through all years of daily records

  for (q in 1:length(csvfiles)){

    daily[[q]] <- read_csv(csvfiles[[q]], col_names = TRUE)
    subdat <- as.data.table(daily[[q]])
    daily[[q]] <- 0

    # Summarize by Station the minimum and maximum operation dates
    stnmindt <- subdat[, lapply(.SD, min, na.rm=TRUE), .SDcols='date',
                       by=list(StationID, loc, elem, year)]
```

```
  stnmaxdt <- subdat[, lapply(.SD, max, na.rm=TRUE), .SDcols='date',
                     by=list(StationID, loc, elem, year)]
  setnames(stnmindt, 'date', 'mindate')
  setnames(stnmaxdt, 'date', 'maxdate')
  stndates <- as.data.table(full_join(stnmindt, stnmaxdt,
                                      by = c('StationID', 'loc', 'elem', 'year')))

  rm(subdat, stnmindt, stnmaxdt)

  # Collect years into one data frame
  if(q==1){
   stndtall <- stndates
  }
  if(q>1){
    stndtall <- rbind(stndtall, stndates)
  }

  # Remove files to save memory
  rm(stndates)

  # Call garbage can gc() to save memory
  gc()
}

########################################
#       END INNER LOOP (Years)         #
########################################

rm(csvfiles, daily, w, q)

# Continue through code to end

# Sort the records by location, element (for WIND), year and StationID
stndtall <- stndtall[order(loc, elem, StationID, year),]

# Summarize minimum and maximum station operation dates for all combined years
stnminall <- stndtall[, lapply(.SD, min, na.rm=TRUE), .SDcols='mindate',
                      by=list(StationID, loc, elem)]
stnmaxall <- stndtall[, lapply(.SD, max, na.rm=TRUE), .SDcols='maxdate',
                      by=list(StationID, loc, elem)]
stndtsum <- as.data.table(full_join(stnminall, stnmaxall,
                      by = c('StationID', 'loc', 'elem')))

rm(stnminall, stnmaxall)

# -------- create additional date fields -----------
stndtsum[,minyear := as.integer(substring(mindate,1,4))]
stndtsum[,minmo := as.integer(substring(mindate,5,6))]
stndtsum[,maxyear := as.integer(substring(maxdate,1,4))]
stndtsum[,maxmo := as.integer(substring(maxdate,5,6))]

stndtinv <- as.data.table(right_join(stnloc, stndtsum,
                                     by = c('StationID', 'loc')))

# --------- partial month adjustments ----------------
stndtinv[, daysmaxmo := days_in_month(as.Date(paste(maxyear, maxmo, 15, sep ="-")))]
stndtinv[, begminmo := as.integer(paste0(minyear, ifelse(minmo < 10, "0", ""),
                                        minmo, '01'))]
stndtinv[, endmaxmo := as.integer(paste0(maxyear, ifelse(maxmo < 10, "0", ""),
                                        maxmo, daysmaxmo))]
stndtinv[, clsdmbeg := (mindate - begminmo)]
```

```
stndtinv[, clsdmend := (endmaxmo - maxdate)]
# eliminate calculation fields
stndtinv[ ,':='(daysmaxmo = NULL, begminmo = NULL, endmaxmo = NULL)]
# Compare minimum date with ghcnd-inventory station open year
stndtinv[, clsdmbeg := ifelse(Open < minyear, 0, clsdmbeg)]
stndtinv[, clsdmend := ifelse(Close > maxyear, 0, clsdmend)]
stndtinv[, clsdinm := (clsdmbeg + clsdmend)]

# ---------- full month adjustments ----------------
# ....... Set up count of days in full months before and after station in operation .....
# start with days in months for 'regular' year (non leap year) - then add leap year adj
dys <- days_in_month(as.Date(paste(1961, seq(1:12), 15, sep ="-")))
dysleap <- c(dys, 1)
repdys <- rep.row(dysleap, nrow(stndtinv))
repmos <- rep.row(seq(1:12), nrow(stndtinv))
# ......................................................................................
# Count mins
output <- matrix(0, nrow(stndtinv), 13)
for(i in 1:nrow(stndtinv)){
  output[i,1:12] <- (repmos[i,1:12] < stndtinv$minmo[i])
}
# Adjust days in February for leap years: output[,2] is 0 or 1 for second month February
output[,13] <- leap_year(stndtinv$minyear)*(output[,2])
daysout <- repdys*output
stndtinv$clsdbef <- apply(daysout, 1, sum)
# Count maxs
output <- matrix(0, nrow(stndtinv), 13)
for(i in 1:nrow(stndtinv)){
  output[i,1:12] <- (repmos[i,1:12] > stndtinv$maxmo[i])
}
# Adjust days in February for leap years: output[,2] is 0 or 1 for second month February
output[,13] <- leap_year(stndtinv$maxyear)*(output[,2])
daysout <- repdys*output
stndtinv$clsdaft <- apply(daysout, 1, sum)
# Compare minimum date with ghcnd-inventory station open year
stndtinv[, clsdbef := ifelse(Open < minyear, 0, clsdbef)]
stndtinv[, clsdaft := ifelse(Close > maxyear, 0, clsdaft)]
stndtinv[, clsdfulm := (clsdbef + clsdaft)]

# ------- count base and recent years -------------
stndtall[, bsyrct:= ifelse(year >= BegBsYr & year <= EndBsYr, 1, 0)]
stndtall[, rcyrct:= ifelse(year > EndBsYr, 1, 0)]
stnyrct <- stndtall[, lapply(.SD, sum, na.rm=TRUE), .SDcols=c('bsyrct', 'rcyrct'),
                    by=list(StationID, loc, elem)]
stnyrct[ ,bryrct := (bsyrct + rcyrct)]
stndtfin <- as.data.table(full_join(stndtinv, stnyrct,
                     by = c('StationID', 'loc', 'elem')))

rm(stndtall, stndtsum, stndtinv, stnyrct)
rm(dys, dysleap, daysout, repdys, repmos, output, i)

# Sort the records by location, element (for WIND) and StationID
stndtfin <- stndtfin[order(loc, elem, StationID),]


# Fill in missing State (St) from StationID - valid for recent years ID naming convention
stndtfin$St <- ifelse(is.na(stndtfin$St), as.character(substring(stndtfin$StationID, 4, 5)),
                      as.character(stndtfin$St))
# If State (St) filled in from StationID, still missing lat, lon, and elev (1998-2016)

attach(stndtfin)
```

```
  stndtfin$begbs <- ifelse(minyear<=BegBsYr & maxyear>=BegBsYr, BegBsYr,
                           ifelse(maxyear<BegBsYr, 0, ifelse(minyear>EndBsYr, 0, minyear)))
  stndtfin$endbs <- ifelse(minyear<=EndBsYr & maxyear>=EndBsYr, EndBsYr,
                           ifelse(minyear>EndBsYr, 0, ifelse(maxyear<BegBsYr, 0, maxyear)))
  stndtfin$begrc <- ifelse(maxyear<=EndBsYr, 0, ifelse(minyear <=EndBsYr+1,
                                                        EndBsYr+1, minyear))
  stndtfin$endrc <- ifelse(maxyear<=EndBsYr, 0, maxyear)
  attach(stndtfin)
  stndtfin$bsspan <- ifelse((begbs == 0 | endbs == 0), 0, endbs - begbs + 1)
  stndtfin$rcspan <- ifelse((begrc == 0 | endrc == 0), 0, endrc - begrc + 1)
  stndtfin$brspan <- stndtfin$bsspan + stndtfin$rcspan
  # stndtfin$spanyrs <- maxyear - minyear + 1
  # eliminate calculation fields
  stndtfin[ ,':='(begbs = NULL, endbs = NULL, begrc = NULL, endrc = NULL)]
  # Name files to write station inventories
  filenmstns <- paste0("_USCANstndt", SelElem, "df.csv")
  # Write station inventories to csv files
  setwd(diroutput)
  write_csv(stndtfin, filenmstns, col_names=TRUE)

  rm(stndtsum, stndtfin, stndtinv, stndtall)
  rm(filenmstns, csvfiles)

  gc()  # Call garbage can to spare memory

}
########################################
#     END OUTER LOOP (Elements)        #
########################################
rm(USCANstn, USCANinv0, USCANinv, stninv, stnloc)
# rm(BegBsYr, EndBsYr)


##### END Program Code


# ============================================================================================ #
# ===== CODE 3 ========= INITIAL YEAR MONTH SUMMARY (from Daily files) ===================== #
# ============================================================================================ #

# Go back to repeat SETUP at top if R has been closed.

# # Set working directory to access output of daily csv files
setwd(diroutput)

Elements <- c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN', 'WIND')

########################################
#        BEGIN OUTER LOOP              #
########################################

# OUTER LOOP: Loop through list of selected weather elements

for (z in 1:length(Elements)){

  SelElem <- Elements[z]
  # creates the list of all the csv files in the directory
  csvfiles <- dir(pattern = paste0("USCANday",  SelElem, "*"))
  csvfiles   # View list of selected file names
  #csvfiles <- csvfiles[58]  # Select years
```

```
# Define list for inner loop
daily <- list()
########################################
#        BEGIN INNER LOOP              #
########################################

# INNER LOOP: for a given element, loop through all years of daily records

for (q in 1:length(csvfiles)){

  daily[[q]] <- read_csv(csvfiles[[q]], col_names = TRUE)
  subdat <- as.data.table(daily[[q]])
  daily[[q]] <- 0

  # Create additional field columns
  subdat[, VALsqd_US := (VAL_US)^2]
  subdat[, zerrec := (VAL <= 0) + 0]
  subdat[, recs := 1]
  subdat[, zblank := ifelse(MFlag == 'P', 1, 0)]
  subdat[, zerobs := (zerrec - zblank)]
  subdat[, obs := (recs - zblank)]

  # Select columns to summarize based on SelElem (*Change columns*)
  yrmocol <- c('VAL', 'VAL_US', 'VALsqd_US', 'zerrec', 'recs', 'zblank',   'zerobs', 'obs')


  # Aggregate data into monthly summaries
  yrmosum <- subdat[, lapply(.SD, sum, na.rm=TRUE), .SDcols=yrmocol,
                  by=list(StationID, loc, elem, year, month)]
  setnames(yrmosum, "VAL", "VALm")
  setnames(yrmosum, "VAL_US", "VALm_US")
  setnames(yrmosum, "VALsqd_US", "sumVALsqd_US")
  # Add fields and reorganize columns
  firstcols <- yrmosum[, StationID:sumVALsqd_US]  # reorganize columns
  firstcols[,VALsqm_US := (VALm_US)^2]
  lastcols <- yrmosum[,zerrec:obs]                # reorganize columns
  yrmosum <- cbind(firstcols, lastcols)
  yrmosum[,daysinmo := days_in_month(as.Date(paste(yrmosum$year,
                                        yrmosum$month, 15, sep ="-")))]
  yrmosum[,misclsd := (daysinmo - obs)]
  rm(firstcols, lastcols)

  #yrmosum[,MEANd_US:= VALm_US/obs]
  rm(subdat)

  rm(yrmocol)

  # Collect years into one data frame
  if(q==1){
    yrmoall <- yrmosum
  }
  if(q>1){
    yrmoall <- rbind(yrmoall, yrmosum)
  }

  # Remove files to save memory
  rm(yrmosum)

  # Call garbage can gc() to save memory
  gc()
```

```
  }
  ########################################
  #       END INNER LOOP (Years)         #
  ########################################

  # Sort the files by location and StationID
  yrmoall <- yrmoall[order(elem, year, StationID), ]

  # Name files to write all years of monthly summarized data out to
  filenmyrmo <- paste0("_USCANyrmo0", SelElem, "df.csv")
  # Write MONTHLY subsets of data to csv files
  write_csv(yrmoall, filenmyrmo, col_names=TRUE)

  rm(yrmoall)
  rm(filenmyrmo)
  rm(csvfiles)

}

########################################
#     END OUTER LOOP (Elements)        #
########################################

# Clear variables to move on to next code
rm(daily, q, z, SelElem)

#### End program code


# ========================================================================================== #
# ===== CODE 4 ====== MERGE STATION INVENTORIES/LOCATIONS TO YEAR MONTH SUMMARY =========== #
# ========================================================================================== #
# ================================ SUMMARIZE BY YEAR ===================================== #
# ========================================================================================== #

# Go back to repeat SETUP at top if R has been closed.

# Set working directory to access output of initial year-month summary and station inventory
setwd(diroutput)

Elements <- c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN', 'WIND')
# Elements <- Elements[1:2] # Select elements (option)

########################################
#        BEGIN SINGLE LOOP             #
########################################

# Loop through list of selected weather elements

for (j in 1:length(Elements))
{
  SelElem <- Elements[j]

  # Find file name to read in - year month data summary
  fileyrmo <- paste0("_USCANyrmo0", SelElem, "*")
  yrmofiles <- dir(pattern = fileyrmo) # creates the list of the files in the directory
  yrmofiles   # view files selected

  # Find file name to read in - station dates of operation and inventories
  filenmstn <- paste0("_USCANstndt", SelElem, "*")
  stnfiles <- dir(pattern = filenmstn) # creates the list of the files in the directory
```

```
stnfiles   # view files selected

# Read in Data Files - yearly station data, station dates of operation, station locations
yrmodat <- read_csv(yrmofiles[1], col_names = TRUE)
stndates <- read_csv(stnfiles[1], col_names = TRUE)

# Remove variables to clean up environment
rm(fileyrmo, filenmstn, yrmofiles, stnfiles)

# Create data tables to process faster
yrmodat <- as.data.table(yrmodat)
stndates <- as.data.table(stndates)

# ---------- merge partial month adj to 'Year Month' data ----------------
stnmins <- stndates[, .(StationID, elem, minyear, minmo, clsdmbeg)]  #(*Change columns*)
stnmins <- stnmins[clsdmbeg > 0]
setnames(stnmins, 'minyear', 'year')
setnames(stnmins, 'minmo', 'month')

stnmaxs <- stndates[, .(StationID, elem, maxyear, maxmo, clsdmend)] #(*Change columns*)
stnmaxs <- stnmaxs[clsdmend > 0]
setnames(stnmaxs, 'maxyear', 'year')
setnames(stnmaxs, 'maxmo', 'month')

yrmomin <- as.data.table(left_join(yrmodat, stnmins,
                         by = c('StationID', 'elem', 'year', 'month')))
yrmofin <- as.data.table(left_join(yrmomin, stnmaxs,
                         by = c('StationID', 'elem', 'year', 'month')))
yrmofin[is.na(yrmofin)]<- 0
rm(stnmins, stnmaxs, yrmomin)
rm(yrmodat)

# _______ MERGE STATES / PROVINCES / COORDINATES / STATION MIN/MAX DATES __________________

# Add state, coordinates, range of station operation dates (min and max)
yrmodet <-  as.data.table(right_join(stndates[, StationID:maxdate], yrmofin,
                          by = c('StationID', 'loc', 'elem')))  #(*Change Columns*)
# keep yrmofin for yearly summary
# yrmosumdet <- yrmodet[,c(1:2, 21:26, 3:20)]  #(*Change Columns*)
# yrmosumdet <- cbind(yrmodet[,StationID:loc], yrmodet[,St:maxdate],
#                     yrmodet[,elem:clsdmend]) #misinm]) #(*Change Columns*)
yrmodet$clsdmend <- as.integer(yrmodet$clsdmend)
yrmodet$clsdmbeg <- as.integer(yrmodet$clsdmbeg)
yrmodet[, misinm := misclsd - (clsdmbeg + clsdmend)]
yrmodet[, clsdinm := (clsdmbeg + clsdmend)]
# Remove columns
yrmodet[ ,':=' (clsdmbeg = NULL, clsdmend = NULL, misclsd = NULL)]

# Name file to write data out to
fileyrmoinv <- paste0("_USCANyrmoinv", SelElem, "df.csv")
# Write YEAR MONTH inventories of data to csv files
write_csv(yrmodet, fileyrmoinv, col_names=TRUE)
rm(yrmodet, fileyrmoinv)

# Continue to sum by year, using year-month data prior to station location merge
# Add count field for months
yrmofin[, count := 1]
yrmofin$clsdmbeg <- as.integer(yrmofin$clsdmbeg)
yrmofin$clsdmend <- as.integer(yrmofin$clsdmend)
# Select data columns to summarize yearly based on SelElem. Eliminate month field
yrcol <- names(yrmofin)[c(6:19)]   #(*Change columns)
```

```
yrsum <- yrmofin[, lapply(.SD, sum, na.rm=TRUE), .SDcols=yrcol,
                 by=list(StationID, loc, elem, year)]
setnames(yrsum, "VALm", "VALy")
setnames(yrsum, "VALm_US", "VALy_US")
setnames(yrsum, "VALsqm_US", "sumVALsqm_US")
setnames(yrsum, "daysinmo", "daysum")
setnames(yrsum, 'count', 'monthct')

# Rearrange column order and insert new field columns
firstcols <- yrsum[,StationID:sumVALsqm_US]
firstcols[,VALsqy_US := VALy_US^2]
midcols <- yrsum[,zerrec:daysum]
midcols[,daysinyr := 365+leap_year(yrsum$year)*1]
yrsum <- cbind(firstcols, midcols, yrsum[,misclsd:monthct])
# Sort data
yrsum <- yrsum[order(StationID, elem, year),]
# Remove data sets to free space
rm(firstcols, midcols, yrcol)
# Keep yrmofin to spread monthly observations to a yearly format

# ------ merge full month adjustments to 'Year' data ----------------
stnyrmins <- stndates[, c("StationID", "elem", "minyear", "clsdbef")]    #(*Change Columns*)
stnyrmins <- stnyrmins[clsdbef > 0]
setnames(stnyrmins, 'minyear', 'year')

stnyrmaxs <- stndates[, c("StationID", "elem", "maxyear", "clsdaft")]    #(*Change Columns*)
stnyrmaxs <- stnyrmaxs[clsdaft > 0]
setnames(stnyrmaxs, 'maxyear', 'year')

yrmin <- as.data.table(left_join(yrsum, stnyrmins,
                        by = c('StationID', 'elem', 'year')))
yrfin <- as.data.table(left_join(yrmin, stnyrmaxs,
                        by = c('StationID', 'elem', 'year')))
rm(stnyrmins, stnyrmaxs, yrmin)
rm(yrsum)
yrfin[is.na(yrfin)]<- 0

yrfin[, clsdinm := (clsdmbeg + clsdmend)]
yrfin[, clsdfulm := (clsdbef + clsdaft)]
yrfin[, clsdall := clsdinm + clsdfulm]

yrfin[, misinm := (misclsd - clsdinm)]
yrfin[, misfulm := (daysinyr - daysum - clsdfulm)]
yrfin[, misall := (misinm + misfulm)]

# Reduce columns
yrfin[ ,':=' (clsdmbeg = NULL, clsdmend = NULL)]
yrfin[ ,':=' (clsdbef = NULL, clsdaft = NULL)]
yrfin[ , misclsd := NULL]

# Spread monthly observations in a 12 column grid for each station - year
# yrmoobs <- yrmofin[, .(StationID, elem, year, month, obs)]
# yrmogrid <- spread(yrmoobs, month, obs)
# yrmogrid[is.na(yrmogrid)] <- 0
# rm(yrmoobs)
# colnames(yrmogrid) <- c('StationID', 'elem', 'year',  'mo01',  'mo02', 'mo03', 'mo04',
# 'mo05', 'mo06', 'mo07', 'mo08', 'mo09', 'mo10', 'mo11',  'mo12')
# Compare to 'obs' data check
# yrmogrid[, yrobs := mo01+mo02+mo03+mo04+mo05+mo06+mo07+mo08+mo09+mo10+mo11+mo12]
# yrmogrid <- yrmogrid[order(StationID, elem, year),]
```

```
  #yrsumgrid <- as.data.table(left_join(yrfin, yrmogrid, by = c('StationID', 'elem', 'year')))
  #rm(yrfin, yrmogrid)

  stndat <- cbind(stndates[,bsyrct:brspan], stndates[,StationID:maxdate])
  rm(stndates)
  yrdet <- as.data.table(left_join(yrfin, stndat, by = c('StationID', 'loc', 'elem')))
  rm(yrfin, stndat)
  #rm(stndates)

  # Reorder data to organize for output
  #(*Change Columns*)
  yrsumdet <- cbind(yrdet[,StationID:loc], yrdet[,St:maxdate], yrdet[,elem:brspan])
  rm(yrdet)

  # Sort data
  yrsumdet <- yrsumdet[order(elem, year, StationID), ]

  # Name file to write data out to
  fileyrdet <- paste0("_USCANyrinvgrid", SelElem, "df.csv")
  # Write YEARLY summary to csv files
  write_csv(yrsumdet, fileyrdet, col_names=TRUE)
  rm(fileyrdet, yrsumdet)

  gc()

}
########################################
#         END LOOP (Elements)          #
########################################

##### End Program Code


# ============================================================================================ #
# ===== CODE 5 ============== Missing Records Summary by Year (loc) ======================== #
# ============================================================================================ #

# Go back to repeat SETUP at top if R has been closed.

# Set working directory to access initial year-month summary and station inventory
setwd(diroutput)

########################################
#            BEGIN LOOP                #
########################################

# Loop through yearly inventories, sum all stations by year

yrfiles <- dir(pattern = "_USCANyrinvgrid") # creates the list of the files in the directory
yrfiles  # view files selected

for (g in 1:length(yrfiles)){

  # Read in Data Files - yearly station data, station dates of operation, station locations
  yrdat <- read_csv(yrfiles[g], col_names = TRUE)
  yrdat <- as.data.table(yrdat)
  # Reduce columns
  yrdat <- yrdat[, StationID:misall]  # Select consecutive columns

  ###################################################
  # Summarize missing records by year and loc
```

```
  yrdat[,stnct := 1]
  miscols <- names(yrdat)[c(18:32)]   #(*Change columns*)
  misrecsum <- yrdat[, lapply(.SD, sum, na.rm=TRUE), .SDcols=miscols,
                      by=list(loc, elem, year)]
  misrecsum <- as.data.table(misrecsum)
  misrecsum <- misrecsum[order(loc, elem, year),]


  if (g==1){
    misrecall <- misrecsum
  }
  if (g > 1){
    misrecall <- rbind(misrecall, misrecsum)
  }

  rm(misrecsum, yrdat)
}

# Define percentage fields (Note zerobs-to-obs has a different denom)
misrecall[, pctobsyr := (obs/daysinyr)]
misrecall[, pctmisyr := (misall/daysinyr)]
misrecall[, pctclsdyr := (clsdall/daysinyr)]
misrecall[, pctzblkyr := (zblank/daysinyr)]
misrecall[, pctzeroyr := (zerobs/daysinyr)]
misrecall[, pctzerobs := (zerobs/obs)]
setnames(misrecall, 'daysum', 'stndays')
setnames(misrecall, 'monthct', 'stnmos')
setnames(misrecall, 'daysinyr', 'stndysinyr')

# Rearrange column order
misrecfin <- cbind(misrecall[, loc:year], misrecall[, pctzerobs:pctobsyr],
                   misrecall[, .(stnct)], misrecall[, zerrec:misall])


#misrecfin[, daysyravg := stndysinyr/stns] # check it is 365 or 366
misrecfin[, obsavg := (obs/stnct)]
misrecfin[, clsdavg := (clsdall/stnct)]
misrecfin[, misavg := (misall/stnct)]
misrecfin[, zblankavg := (zblank/stnct)]
misrecfin[, zeroavg := (zerobs/stnct)]
misrecfin[, mosavg := (stnmos/stnct)]


########################################
#         END LOOP (Elements)          #
########################################

# Write out Missing Record Summary for All Elements
write_csv(misrecall, "_USCANmisrecELEMdf.csv", col_names=TRUE)

# Reopen option
# misrecall <- read_csv("_USCANmisrecELEMdf.csv", col_names=TRUE)
# misrecall <- as.data.table(misrecall)

mistbl <- misrecall[,c('loc', 'elem', 'year', 'pctmisyr')]
clsdtbl <- misrecall[,c('loc', 'elem', 'year', 'pctclsdyr')]
obstbl <- misrecall[,c('loc', 'elem', 'year', 'pctobsyr')]
stntbl <- misrecall[,c('loc', 'elem', 'year', 'stnct')]
zerobstbl <- misrecall[,c('loc', 'elem', 'year', 'pctzerobs')]

mistbl <- mistbl[elem %in% c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN')]
```

```
mistblw <- spread(mistbl, elem, pctmisyr)
mistblw[is.na(mistblw)] <- ""
mistblw

clsdtbl <- clsdtbl[elem %in% c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN')]
clsdtblw <- spread(clsdtbl, elem, pctclsdyr)
clsdtblw[is.na(clsdtblw)] <- ""
clsdtblw

obstbl <- obstbl[elem %in% c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN')]
obstblw <- spread(obstbl, elem, pctobsyr)
obstblw[is.na(obstblw)] <- ""
obstblw

zerobstbl <- zerobstbl[elem %in% c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN')]
zerobsw <- spread(zerobstbl, elem, pctzerobs)
zerobsw[is.na(zerobsw)] <- ""
zerobsw

write_csv(obstblw, 'pctmisrec.csv', col_names=TRUE)
write_csv(mistblw, 'pctmisrec.csv', col_names=TRUE)
write_csv(clsdtblw, 'pctclsdrec.csv', col_names=TRUE)
write_csv(zerobsw,  'pctzerobs.csv', col_names=TRUE)

##### END PROGRAM CODE



# ========================================================================================== #
# ===== CODE 6 ========== BIMONTHLY (MEI) and TRIMONTHLY (ONI) INDICES ===================== #
# ========================================================================================== #

# Go back to repeat SETUP at top if R has been closed.

Elements <- c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN', 'WIND')

# Open station level summaries by year month; sum by two- and three- month periods.
# The MultiVariate ENSO Index (MEI) is bimonthly (two months).
# The Oceanic Nino Index (ONI) is trimonthly (three months).

# Read in MEI and ONI from base directory
setwd(dirbase)
MEI <- read_csv('MEI_Index.csv', col_names = TRUE)
colnames(MEI) <- c('year', 'DecJan', 'JanFeb', 'FebMar', 'MarApr', 'AprMay', 'MayJun',
                    'JunJul',  'JulAug',  'AugSep', 'SepOct', 'OctNov', 'NovDec')
ONI <- read_csv('ONI_Index.csv', , col_names = TRUE)
head(ONI)

# use functions from tidyvers package to convert index lists from wide to long
MEIlong <- gather(MEI, "bimo", "MEI", 2:13)
head(MEIlong)
ONIlong <- gather(ONI, "trimo", "ONI", 2:13)
head(ONIlong, 10)
setnames(ONIlong, 'Year', 'year')
rm(MEI, ONI)

# Create labels for bimonthly and trimonthly aggregations
bimolabel <- data.frame(k = seq(1, 12, 1),
                        bimo = c('JanFeb','FebMar', 'MarApr', 'AprMay', 'MayJun', 'JunJul',
                                 'JulAug', 'AugSep', 'SepOct', 'OctNov', 'NovDec', 'DecJan'))
trimolabel <- data.frame(k = seq(1, 12, 1),
```

```
                        trimo = c('JFM','FMA', 'MAM', 'AMJ', 'MJJ', 'JJA', 'JAS', 'ASO',
                        'SON', 'OND', 'NDJ', 'DJF'))


# Set working directory to access year-month summaries
setwd(diroutput)
########################################
#          BEGIN MAIN LOOP            #
########################################

for (h in 1:length(Elements))
{
  SelElem <- Elements[h]

  fileyrmo <- paste0("_USCANyrmoinv", SelElem, "df.csv")
  fileyrmo
  filestndt <- paste0("_USCANstndt", SelElem, "df.csv")
  filestndt
  #  yrmofiles <- dir(pattern = fileyrmo) # creates the list of the files in the directory
  #  yrmofiles  # view files selected

  setwd(diroutput)
  # Read in Data files - year month data with missing records and closed dates inventories
  yrmoall <- read_csv(fileyrmo[1], col_names = TRUE)
  # Select columns and place in data table for efficiency
  yrmodat <- as.data.table(yrmoall)   # 8,579,187
  rm(yrmoall, fileyrmo)
  minyr <- min(yrmodat$year)
  maxyr <- max(yrmodat$year)

  yrmodat[, monthct := 1]   # Count months summed in loop
  yrmodat$monthX <- yrmodat$month
  # Create 'Month 13' data as January of next year, 'Month 14' as February of next year
  wrapmo13 <- yrmodat[month == 1]
  wrapmo14 <- yrmodat[month == 2]
  wrapmo13$monthX <- 13
  wrapmo13$year <- wrapmo13$year - 1
  wrapmo14$monthX <- 14
  wrapmo14$year <- wrapmo14$year - 1
  loopsum <- rbind(yrmodat, wrapmo13, wrapmo14)
  loopsum <- loopsum[year >= minyr]
  loopsum <- loopsum[order(elem, StationID, year, monthX), ]
  rm(yrmodat)

  # Create lists for loop
  bimosum <- list()
  trimosum <- list()
  # Select columns of values to sum in loop
  sumcols <- names(loopsum)[c(14:26)]   #(*Change columns*)
  ########################################
  #      BEGIN MINOR LOOP               #
  ########################################
  for (k in 1:12){
    mo1 <- as.character(k)
    mo2 <- as.character(k+1)
    mo3 <- as.character(k+2)
    #mo2lab <- ifelse(k == 12, 1, mo2)
    #mo3lab <- ifelse(k == 12, 2, mo3)
    bimosub <- loopsum [monthX == mo1 | monthX == mo2]
    trimosub <- loopsum[monthX %in% c(mo1, mo2, mo3)]
    bimosum[[k]] <- bimosub[, lapply(.SD, sum, na.rm=TRUE), .SDcols=sumcols,
```

```
                         by=list(StationID, loc, elem, year)]
  trimosum[[k]] <- trimosub[, lapply(.SD, sum, na.rm=TRUE), .SDcols=sumcols,
                           by=list(StationID, loc, elem, year)]
  bimosum[[k]]$ord <- k
  trimosum[[k]]$ord <- k
  bimosum[[k]]$bimo = bimolabel[k,2]
  trimosum[[k]]$trimo = trimolabel[k,2]
}
########################################
#         END MINOR LOOP               #
########################################

# Continue through code to end

rm(wrapmo13, wrapmo14, mo1, mo2, mo3, k)
rm(loopsum, bimosub, trimosub, sumcols)

bimosumall <- rbindlist(bimosum)
trimosumall <- rbindlist(trimosum)
setnames(bimosumall, "daysinmo", "daysum")
setnames(trimosumall, "daysinmo", "daysum")
setnames(bimosumall, "VALm", "VAL2m")
setnames(trimosumall, "VALm", "VAL3m")
setnames(bimosumall, "VALm_US", "VAL2m_US")
setnames(trimosumall, "VALm_US", "VAL3m_US")
setnames(bimosumall, "VALsqm_US", "sumVALsqm_US")
setnames(trimosumall, "VALsqm_US", "sumVALsqm_US")
rm(bimosum, trimosum)

class(bimosumall$year)
bimosumall$delete <- ifelse((bimosumall$year == maxyr & bimosumall$ord == 12), 'Y', 'N')
trimosumall$delete <- ifelse((trimosumall$year == maxyr & trimosumall$ord == 11), 'Y',
                      ifelse((trimosumall$year == maxyr & trimosumall$ord == 12), 'Y', 'N'))

bimofin <- subset(bimosumall, bimosumall$delete == 'N')
trimofin <- subset(trimosumall, trimosumall$delete == 'N')
bimofin[, delete:=NULL]
trimofin[, delete:=NULL]
rm(bimosumall, trimosumall)
#nrow(bimosumfin)  # 8,750,322 PRCP
#nrow(trimosumfin) # 8,846,206 PRP
bimofin$year <- as.integer(bimofin$year)
bimofin$bimo <- as.character(bimofin$bimo)
trimofin$year <- as.integer(trimofin$year)
trimofin$trimo <- as.character(trimofin$trimo)

# Merge MEI (bimo) and ONI (trimo) indices with summed data
bimoInd <- as.data.table(left_join(bimofin, MEIlong,
                          by = c('year', 'bimo')))
trimoInd <- as.data.table(left_join(trimofin, ONIlong,
                           by = c('year', 'trimo')))
rm(bimofin, trimofin)

# Merge Station inventories / states / coordinates
setwd(diroutput)
stndates <- read_csv(filestndt, col_names = TRUE)
stndates  <- as.data.table(stndates)

bimoIndSt <- as.data.table(right_join(stndates[, StationID:maxdate], bimoInd,
                            by = c('StationID', 'loc', 'elem'))) #(*Change columns*)#
rm(bimoInd)
```

```
  trimoIndSt <- as.data.table(right_join(stndates[, StationID:maxdate], trimoInd,
                              by = c('StationID', 'loc', 'elem'))) #(*Change columns*)#
  rm(trimoInd)
  rm(stndates)

  setwd(diroutput)
  bimofilenm <- paste0('_USCANbimo', SelElem, 'df.csv')
  trimofilenm <- paste0('_USCANtrimo', SelElem, 'df.csv')
  write_csv(bimoIndSt, bimofilenm, col_names = TRUE)
  write_csv(trimoIndSt, trimofilenm, col_names = TRUE)

  rm(bimofilenm, trimofilenm, filestndt)
  rm(bimoIndSt, trimoIndSt)

}

########################################
#       END MAIN LOOP (Elements)       #
########################################

rm(bimolabel, trimolabel, MEIlong, ONIlong)
rm(h, maxyr, minyr)

##### End Program Code



# ========================================================================================== #
# ===== CODE 7 ========================== WEEKLY INDICES ================================== #
# ========================================================================================== #

# Go back to repeat SETUP at top if R has been closed.

Elements <- c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN', 'WIND')

# For weekly Nino Indices choose BegDat <- "1989/12/31"
# (Weekly SST data starts week centered on 1990/01/03)
# -----------------------------------------------------
BegDate <- "1961/01/01"    # 1961/01/01 gives same grouping as 1989/12/31
EndDate <- "2017/12/31"
# --------- weekly labels for grouping sums -----------
dte = seq(as.Date(BegDate), as.Date(EndDate), "days")  # sequence of dates from beg to end
nodys <- length(dte)   # Number of days in sequence
noweeks <- as.integer(nodys/7)   # Number of full weeks in sequence
remdys <- nodys%%7             # Remaining days, not full weeks
weeklabel <- data.frame(weekno = c(rep(1:noweeks, each = 7), rep(noweeks+1, remdys)), dte =
seq(as.Date(BegDate), as.Date(EndDate), "days"))
weeklabel$chardate <- paste0(as.character(substring(weeklabel$dte, 1, 4)),
as.character(substring(weeklabel$dte, 6, 7)), as.character(substring(weeklabel$dte, 9,10)))
weeklabel$date <- as.integer(weeklabel$chardate)
startlab <- as.Date(BegDate)+3      # Weeks are labeled by middle day
centerdates <- seq(as.Date(startlab), by = "7 days", length.out = noweeks)  # middle day for
each week of sequence
weeklabel$ctrweek <- c(rep(centerdates, each = 7), rep(as.Date(tail(centerdates,1))+7,
remdys))  # label all days
weeklabel$yrgrp <- substring(weeklabel$ctrweek, 1, 4)
weeklabel <- weeklabel[,c(1,4:6)]
rm(BegDate, EndDate, centerdates, dte, nodys, remdys, startlab, noweeks)
# -----------------------------------------------------

########################################
```

```
#          BEGIN OUTER LOOP            #
########################################

for (z in 1:length(Elements)){

  # Set working directory to access output of daily csv files
  setwd(diroutput)

  SelElem <- Elements[z]
  # creates the list of all the csv files in the directory
  csvfiles <- dir(pattern = paste0("USCANday",  SelElem, "*"))
  csvfiles <- csvfiles[30:58]  # select years since 1989 to match Nino indices
  # csvfiles

  daily <- list()
  ########################################
  #    BEGIN FIRST INNER LOOP            #
  ########################################
  # for selected element, loop through all years of daily records

  for (q in 1:length(csvfiles)){

    # Set working directory to access output of daily csv files
    setwd(diroutput)

    daily[[q]] <- read_csv(csvfiles[[q]], col_names = TRUE)
    subdat <- as.data.table(daily[[q]])
    daily[[q]] <- 0

    # Create additional field columns
    subdat[, VALsqd_US := (VAL_US)^2]
    subdat[, zerrec:= (VAL <= 0) + 0]
    subdat[, recs:= 1]
    subdat[, zblank := ifelse(MFlag == 'P', 1, 0)]
    subdat[, zerobs:= (zerrec - zblank)]
    subdat[, obs := (recs - zblank)]

    # Include additional week labels for summarizing by week (already have year and month)
    subdatwk <- as.data.table(left_join(subdat, weeklabel, by = 'date'))
    rm(subdat)

    # Select columns to summarize based on SelElem
    wkcol <- names(subdatwk)[c(4, 13:19)]   #(*Change columns)

    weeksum <- subdatwk[, lapply(.SD, sum, na.rm=TRUE), .SDcols=wkcol,
                         by=list(StationID, loc, elem, weekno, yrgrp, ctrweek)]
    setnames(weeksum, 'VAL', 'VALw')
    setnames(weeksum, 'VAL_US', 'VALw_US')
    firstcols <- weeksum[, StationID:VALsqd_US]
    firstcols[, VALsqw_US := (VALw_US)^2]    # PRCP 3,339 rows = 64 States x 52 weeks roughly
    weeksum <- cbind(firstcols, weeksum[, zerrec:obs])

    rm(firstcols)
    rm(subdatwk)
    rm(wkcol)

    # Sort columns
    weeksum <- weeksum[order(elem, loc, weekno),]

    if(q == 1){
      weeksall <- weeksum
```

```
  }
  if( q > 1){
    weeksall <- rbind(weeksall, weeksum)
  }

  rm(weeksum)

  gc()  # Call for garbage can to spare memory

}

########################################
#     END FIRST INNER LOOP (Years)     #
########################################

rm(daily, csvfiles, q)

# Weeks numbered according to Nino indices overlap years;
# Sum again to complete summation of boundary weeks.
wkscol <- names(weeksall)[7:15]  #(*Change columns)
weekly <- weeksall[, lapply(.SD, sum, na.rm=TRUE), .SDcols=wkscol,
                    by=list(StationID, loc, elem, weekno, yrgrp, ctrweek)]
rm(weeksall)  #PRCP sum removes 306,156 rows
rm(wkscol)
# Calculate missing or closed records based on complete 7-day week sums
weekly[, misclsd := (7 - obs)]

gc()


# Read in list of stations with state/province, elevation, coordinates
setwd(dirbase)
stnlist <- read_csv('ghcnd-stations.csv', col_names = TRUE)
colnames(stnlist) <- c("StationID","lat", "lon", "elev", "St", "Name", "GSNFlag", "zip" )
stnlist$loc <- as.character(substring(stnlist$StationID, 1,2))
stnlist$StationID <- as.character(stnlist$StationID)
stnsub <- stnlist[,c(1:5,9)]  #(*Change columns*)
USCANstn <- subset(stnsub, stnsub$loc %in% c('US', 'CA'))
rm(stnlist, stnsub)

# Read in Nino Indices
setwd(dirbase)
ninoweekly <- read_csv('NinoWeekly.csv', col_names = FALSE)
colnames(ninoweekly) <- c("ctrweek","Nino12Ind", "Nino12Anom", "Nino3Ind", "Nino3Anom",
                           "Nino34Ind", "Nino34Anom", "Nino4Ind", "Nino4Anom")


########################################
#   BEGIN SECOND INNER LOOP (Years)    #
########################################
# Select data year by year and merge with station locations and weekly Nino indices
for  (p in 1:length(unique(weekly$yrgrp))){
  yrwrite <-  as.integer(min(weekly$yrgrp)) + (p-1)
  weeksel <- weekly[yrgrp == yrwrite]

  weekst <- as.data.table(left_join(weeksel, USCANstn[,1:5], by = 'StationID'))
  rm(weeksel)
  weekselst <- cbind(weekst[, StationID:elem], weekst[,.(St)], weekst[,lat:elev],
                    weekst[,weekno:misclsd])   #(*Change columns*)
  rm(weekst)
```

```
    # Include weekly Nino Indices from 1990 to present in weekly table
    weekselIndex <- as.data.table(left_join(weekselst, ninoweekly, by = 'ctrweek'))
    rm(weekselst)

    setwd(diroutput)
    filenmweek <- paste0("USCANweek", Elements[z], yrwrite, "df.csv")
    write_csv(weekselIndex, filenmweek, col_names = TRUE)
    rm(yrwrite, weekselIndex)
  }

  ########################################
  #    END SECOND INNER LOOP (Years)     #
  ########################################
  rm(USCANstn, ninoweekly, weekly, filenmweek)

}
########################################
#     END OUTER LOOP (Elements)        #
########################################

rm(p, z, weeklabel)

##### End Program Code


# EXTRA CODE - does not link to weekly indices

# ------- weekly labels for weeks of all years to fall on the same days -----------
# Choose any non-leap year from Jan 1 to Dec 31
BegDate <- "1961/01/01"
EndDate <- "1961/12/31"
dt = seq(as.Date(BegDate), as.Date(EndDate), "days")  # sequence of dates from beg to end
weekno = as.integer((c(rep(1:52, each = 7), 52)))

startlab <- as.Date(BegDate)+3
centerdates <- seq(as.Date(startlab), by = "7 days", length.out = 52)
ctrweek <- c(rep(centerdates, each = 7), tail(centerdates, 1))

weeklab <- data.frame(weekno, ctrweek, dt)
dtleap = as.Date("1964/02/29")  # Choose any leap day
newrow <- data.frame(weekno = as.integer(9), ctrweek = centerdates[9], dt = dtleap)
weeklabel <- rbind(weeklab, newrow)
weeklabel$moday <- format(as.Date(weeklabel$dt), "%m-%d")
weeklabel$mdchar <- as.character(paste0(substring(as.character(weeklabel$moday), 1, 2),
substring(as.character(weeklabel$moday), 4, 5)))
weeklabel <- weeklabel[order(weeklabel$moday),]
weeklabel <- weeklabel[,c(1,5, 2)]
rm(BegDate, EndDate, centerdates, dt, dtdleap, newrow, ctrweek, startlab, weeklab)

# Code to advance year (if merging by full date)
weeklabel$ctrweek <- paste0(as.character(as.integer(substring(weeklabel$ctrweek, 1, 4))+1),
                            substring(weeklabel$ctrweek, 5, 10))

# ================================================================================== #
# ===== CODE 8 ============ SELECT STATIONS and SUMMARIZE BY STATE ======================== #
# ================================================================================== #
# ================================ PLOT STATIONS ===================================== #
# ================================================================================== #
# =============================== MAP STATE CHOROPLETH ================================== #
# ================================================================================== #
```

```
# Go back to repeat SETUP at top if R has been closed.

# Select base years (eg. 1961 - 1990) for climatology, typically 30 past years
BegBsYr <- 1961
EndBsYr <- 1990

# Select weather element to identify summary files
SelElem <- 'PRCP'

# for mapping - read in state names and abbreviations
setwd(dirbase)
stabbr <- read_csv('ghcnd-states.csv', col_names = FALSE)
colnames(stabbr) <- c('St', 'Name')
stabbr$Name <- tolower(stabbr$Name)  # need lower case for package 'maps'

# Set working directory to access output of yearly csv files
setwd(diroutput)
yrfilenm <- paste0('_USCANyrinvgrid', SelElem, 'df.csv')
yrall <- read_csv(yrfilenm, col_names = TRUE)
yrall <- as.data.table(yrall)      #   785,941 PRCP

rm(yrfilenm)

table(yrall$bryrct)
table(yrall$bsyrct)
# Select Stations from yearly data
yrsel <- yrall[loc == 'US' & bryrct == 57]
#yrsel <- yrall[loc == 'US' & bsyrct > 24 & rcyrct > 22]

# OPTION: Plot comparisons - all US stations
library(maps)
USall <- yrall[loc == 'US']
map("state")
points(USall$lon, USall$lat, pch=19, cex = 0.05, col = 'dodgerblue3')
# Plot comparisons - only selected US stations
map("state")
points(yrsel$lon, yrsel$lat, pch=19, cex = 0.05, col = 'dodgerblue3')

# REVIEW SELECTION OF STATIONS
# view count of stations in summary data
length(unique(yrall$StationID))
# view count of stations in selection
length(unique(yrsel$StationID))
# view number of stations selected by State
table(yrsel$St)

# Add field for station count by state or year count by station
yrsel[, ct := 1]
yrselcol <- c( "VALy", "VALy_US", "obs", "ct")  #(*Change columns*)
yrselsum <- yrsel[, lapply(.SD, sum, na.rm=TRUE), .SDcols=yrselcol,
                  by=list(loc, St, elem, year)]
yrstns <-   yrsel[, lapply(.SD, sum, na.rm=TRUE), .SDcols='ct',
                  by=list(StationID, St, lat, lon)]
yrselsum <- as.data.table(yrselsum)
setnames(yrselsum, 'ct', 'stnct')
yrselsum[, VALsqSt_US := (VALy_US)^2] # 56 years x 50 states
nrow(yrselsum)  # 58 years x 50 states = 2900

# Base Years State Level Summary
baseyrs <- yrselsum[year >= BegBsYr  & year <= EndBsYr]
# Calculate mean and stdev stats manually by formula
```

```
basesum <- baseyrs[, lapply(.SD, sum, na.rm=TRUE),
                   .SDcols=c('VALy_US', 'VALsqSt_US', 'stnct', 'obs'),
                   by=list(loc, St, elem)]
basesum <- as.data.table(basesum)
basesum[, bsMean := (VALy_US/30)]
basesum[, bsSD := sqrt((VALsqSt_US - ((bsMean^2) * 30))/29)]
setnames(basesum, 'stnct', 'bsstnyrs')
setnames(basesum, 'obs', 'bsstnobs')

# Use package dplyr to calculate mean and st dev, remove NA's
# Warning: dplyr summarise function will not group if package 'plyr' is loaded
# detach(package:plyr)
  anomaly <- baseyrs %>%
  group_by(loc, St, elem) %>%
  summarise(bsstnobs=sum(obs), bsstnyrs = sum(stnct), bsmean = mean(VALy_US),
            bsstdev = sd(VALy_US), rm.na = TRUE)

# head(anomaly)

# Merge base year statistics with yearly data set
yrstat <- as.data.table(left_join(yrselsum, anomaly,
                         by = c('loc', 'St', 'elem')))
# Calculate anomalies by year and state
yrstat[, anom :=  (VALy_US - bsmean) / bsstdev]

# Merge state names with yearly data set, for mapping
yrstatst <- as.data.table(left_join(yrstat, stabbr, by = 'St'))
nrow(yrstatst)

# Create a file name to describe data output to write
fileyrst <- paste0('USyrstat', SelElem, 'df.csv')
write_csv(yrstatst, fileyrst, col_names = TRUE)

# To reopen the data table
# fileyrst <- paste0('yrstat', SelElem, 'df.csv')
# yrstatst <- read_csv('yrselstatsPRCP56.csv', col_names = TRUE)
# yrstatst <- as.data.table(yrstatst)

#..........................................
#...... Choropleth from Scratch .............
#..........................................

#Load package 'maps' to plot custom choropleth
library(maps)

# Select a year of data
SelYr <- 2016
yrst <- yrstatst[year == SelYr]

# Create breaks for ranges in the standard deviations
yrst$STDEVgrp <- cut(yrst$anom,
                     breaks = c(-Inf,  -1.5,  -1,  -0.5,  0, 0.5,  1,  1.5,  2, 2.5, Inf),
                     labels = c("neg2&-", "neg1.5", "neg1.0", "neg0.5", "pos0.50",
                                "pos1.00", "pos1.5", "pos2.0", "pos2.5", "pos3&+"),
                     right = FALSE)

# Create labels and choose color scheme to match the ranges
STDEVgrp= c("neg2&-", "neg1.5", "neg1.0", "neg0.5",
            "pos0.50", "pos1.00", "pos1.5", "pos2.0", "pos2.5", "pos3&+")
colreg <- c("wheat3", "wheat2", "wheat1",  "lightyellow1", "lightcyan1", "lightskyblue1",
            "skyblue2", "skyblue3", "skyblue4", "midnightblue")
```

```
# Combine group and colors in a data frame
colregdf <- data.frame(STDEVgrp, colreg)

# Merge colors with year data
yrstcolor <- as.data.table(left_join(yrst, colregdf, by = 'STDEVgrp'))
# Remove Alaska and Hawaii to avoid error in plotting 48 states (Check DC, Puerto Rico etc)
yrstplot <- yrstcolor[St != 'AK' & St != 'HI']

# Split data table into segments by regional colors to plot colors
yrstseg <- split(data.frame(yrstplot), yrstplot[, colreg])

# Draw states then add colors for each split in a loop
map("state")
for (r in 1:length(yrstseg)){
  map("state", region=yrstseg[[r]]$Name, interior=F, fill=T, boundary=T,
      col = as.character(yrstseg[[r]]$colreg[1]), add=T)
}

# Indicate font sizes for map
par(ps = 12, cex = 0.8, cex.main = 2.2)
title(paste0("Precipitation Anomalies by State - ", SelYr))
# Add points for the selected station locations
legendtxt <- c("< -1.5", "-1.5 to -1.0", "-1.0 to -0.5", "-0.5 to 0", "0 to +0.50",
               "+0.5 to +1.0", "+1.0 to +1.5", "+1.5 to +2.0", "+2.0 to +2.5", "> +2.5")
par(ps = 16)
legend("bottomright", legendtxt,  horiz = FALSE, fill = colreg)
# Add points for station locations
points(yrstns$lon, yrstns$lat, pch = 19, cex = 0.3, col = 'maroon4')


#...................................................
#........ Choropleth by Package ggplot2 ............
#...................................................

library(ggplot2)
library(fiftystater)

# ggplot2 base choropleth colors (data can be yrst or yrstcolor)

t <- ggplot(yrstcolor, aes(map_id = yrstcolor$Name, fill=yrstcolor$anom)) +
  geom_map(map = fifty_states, colour = 'black') +
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map() + ggtitle(paste0('Precipitation Anomalies by State in ',
                              SelYr, '\n(Base Years ', BegBsYr,  ' - ', EndBsYr, ')')) +
  theme(plot.title=element_text(size = rel(1.5), lineheight = .9,
                               family = 'Times', colour = 'black', hjust = 0.5)) +
  theme(axis.title.x=element_blank())+
  theme(axis.title.y=element_blank())

t + fifty_states_inset_boxes()

# assigning custom choropleth colors centere at zero

q <-   ggplot(yrstcolor, aes(fill = anom, map_id = Name)) +
  geom_map(map = fifty_states, colour = 'black') +
  expand_limits(x = fifty_states$long, y = fifty_states$lat) + coord_map() +
  scale_fill_gradient2(low = "wheat4", mid = "white", midpoint = 0,
                       high = "dodgerblue4", limits = c(-3,3)) +
  ggtitle(paste0('Precipitation Anomalies by State in ', SelYr,
                 '\n(Base Years ', BegBsYr,  ' - ', EndBsYr, ')')) +
  theme(plot.title=element_text(size = rel(1.5), lineheight = .9,
```

```
                                family = 'Times', colour = 'black', hjust = 0.5)) +
  theme(axis.title.x=element_blank())+
  theme(axis.title.y=element_blank())

q + fifty_states_inset_boxes()

##### End Program Code


# ============================================================================================ #
# ===== CODE 9 ================= COMBINE MONTHLY INDICES =================================== #
# ============================================================================================ #

# Set boundaries on data to combine – make sure years exist in the data
MinYr <- 1951
MaxYr <- 2017

# Read in monthly Nino Indices
setwd(dirbase)
ninomonthly <- read_csv('NinoMonthly.csv', col_names = TRUE)
nino <- as.data.table(ninomonthly)
nino
colnames(nino) <- c('year', 'month', "Nino12", "Anom12", "Nino3", "Anom3", "Nino4", "Anom4",
                      "Nino34", "Anom34")
nino
nino <- nino[year >= MinYr & year <= MaxYr]

SOI <- read_csv('SOI_Anom.csv', col_names = FALSE)
EQSOI <- read_csv('EQSOI.csv', col_names = FALSE)
TNI <- read_csv('TNI.csv', col_names = FALSE)
BEST <- read_csv('BEST1mo.csv', col_names = FALSE)
colnames(SOI) <- c('year', seq(1:12))
colnames(EQSOI) <- c('year', seq(1:12))
colnames(TNI) <- c('year', seq(1:12))
colnames(BEST) <- c('year', seq(1:12))

SOIlg <- as.data.table(gather(SOI, "month", "SOI", 2:13))
EQSOIlg <- as.data.table(gather(EQSOI, "month", "EQSOI", 2:13))
TNIlg <- as.data.table(gather(TNI, "month", "TNI", 2:13))
BESTlg <- as.data.table(gather(BEST, "month", "BEST", 2:13))
SOIlg <- SOIlg[year >= MinYr & year <= MaxYr]
EQSOIlg <- EQSOIlg[year >= MinYr & year <= MaxYr]
TNIlg <- TNIlg[year >= MinYr & year <= MaxYr]
BESTlg <-  BESTlg[year >= MinYr & year <= MaxYr]

IndexAll <- cbind(nino, SOIlg[,3], EQSOIlg[,3], BESTlg[,3], TNIlg[,3])

write_csv(IndexAll, 'IndexMonthly.csv', col_names = TRUE)


##### End Program Code


# ============================================================================================ #
# ===== CODE 10 ================= PLOT INDEX TIME SERIES =================================== #
# ============================================================================================ #

# Load packages
library(reshape2)   # To convert Index data from wide to long
library(ggplot2)    # To produce graphs of indices
```

```
setwd(dirbase)
indices <- read_csv('IndexMonthly.csv', col_names = TRUE)
indices <- as.data.table(indices)
indices[indices < -99] <- NA
IndexName <- names(indices)[3:14] #(*Change Columns*)

IndexName  # view index names (column headers)
# Select index number
i = 4
# assign column number of selected index
indcol <- as.integer(i+2)
SelIndex <- names(indices)[indcol]
# Rename column to be plotted
setnames(indices, SelIndex, "PlotIndex")
PlotDat <- cbind(indices[, year:month], indices[, .(PlotIndex)])
PlotDat$positive <- PlotDat$PlotIndex >= 0  # TRUE/FALSE values
ggplot(PlotDat, aes(x=year, y = PlotIndex, fill = positive))  + geom_bar(stat="identity") +
ylab(SelIndex)
# Reset name of column plotted to original index name
setnames(indices, "PlotIndex", SelIndex)

# If error, reset column names
# colnames(indices) <- c(names(indices)[1:2], IndexName)

MEI <- read.csv('MEI_Index.csv', header = TRUE)
colnames(MEI) <- c('Year', 'DecJan', 'JanFeb', 'FebMar', 'MarApr', 'AprMay', 'MayJun',
                   'JunJul', 'JulAug',  'AugSep', 'SepOct', 'OctNov', 'NovDec')
head(MEI)

# Labels used to sort bimo and trimo ascending for time series
molabels <- data.frame(month = seq(1, 12, 1),
                       mo = c('Jan','Feb', 'Mar', 'Apr', 'May', 'Jun',
                              'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'),
                       bimo = c('JanFeb','FebMar', 'MarApr', 'AprMay', 'MayJun', 'JunJul',
                                'JulAug', 'AugSep', 'SepOct', 'OctNov', 'NovDec', 'DecJan'),
                       trimo = c('JFM','FMA', 'MAM', 'AMJ', 'MJJ', 'JJA',
                                 'JAS', 'ASO', 'SON', 'OND', 'NDJ', 'DJF'))

# use tidyvers gather() to convert wide format to long
MEIlong <- gather(MEI, "bimo", "MEI", 2:13)
MEIlong[MEIlong < -99] <- NA
head(MEIlong)

# Create column to identify positie values in order to color code graph plot
MEIlong$posM <- MEIlong$MEI >= 0    # TRUE/FALSE values
MEIlongno <- merge(MEIlong, molabels[,c(1,3)], by = 'bimo', all.MEIlong = TRUE)
MEIlongno <- MEIlongno[order(MEIlongno$Year, MEIlongno$month),]
head(MEIlongno,15)

ggplot(MEIlongno, aes(x=Year, y = MEI, fill = posM)) + geom_bar(stat="identity") + ylab("MEI")

##### End Program Code


# ============================================================================================ #
# ===== CODE 11 ================ PLOT ELEMENT vs. INDEX by STATE ========================== #
# ============================================================================================ #
# Go back to repeat SETUP at top if R has been closed.
library(stringr)  # Converts all capitals to title format

# Base years (eg. 1961 - 1990) for climatology, that agree with data
```

```
BegBsYr <- 1961
EndBsYr <- 1990

Elements <- c('PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN', 'WIND')

# SELECTIONS FOR PLOTS
SelElem <- 'SNOW'    # Four character weather element abbreviation
SelState <- 'MN'     # Two character state abbreviation
SelIndex <- 1        # 1, 2, 3, ..., 12 column order of index in IndexMonthly.csv
SelMonth <- 1        # 1, 2, 3, ..., 12 month
# Select first month of bimonthly/trimonthly sums:
# 1 = 'JanFeb'/'JFM'; 12 = 'DecJan'/'DJF'

MoLabel <- c('January', 'February', 'March', 'April', 'May', 'June', 'July',
             'August', 'September', 'October', 'November', 'December')
Month2 <- ifelse(SelMonth == 12, 1, SelMonth+1)
Month3 <- ifelse(SelMonth == 11, 1, ifelse(SelMonth == 12, 2, SelMonth +2))
SelBiMo <- paste0(substring(MoLabel[SelMonth], 1, 3), substring(MoLabel[Month2], 1, 3))
TriMoSel <- paste0(substring(MoLabel[SelMonth], 1, 1), substring(MoLabel[Month2], 1, 1),
                   substring(MoLabel[Month3], 1, 1))
BiMoLab <- paste(substring(SelBiMo, 1, 3), substring(SelBiMo, 4, 6), sep = ' - ')
TriMoLab <- paste(BiMoLab, substring(MoLabel[Month3], 1, 3), sep = ' - ')
rm(Month2, Month3)
ElemLab <- data.frame(Elem = Elements[1:5],
              ElemName = c('Rainfall', 'Snowfall','Snow Depth',
              'Max. Temperature', 'Min. Temperature'))
# for mapping - read in state names and abbreviations
setwd(dirbase)
stabbr <- read_csv('ghcnd-states.csv', col_names = FALSE)
colnames(stabbr) <- c('St', 'Name')
# read in indices
setwd(dirbase)
indexmo <- read_csv('IndexMonthly.csv', col_names = TRUE)

# Set Working Directory to access csv data files to plot
setwd(diroutput)

#  Read in Yearly Inventory
yrfile <- paste0('_USCANyrinvgrid', SelElem, 'df.csv')
yrdat <- read_csv(yrfile, col_names = TRUE)
yrdat <- as.data.table(yrdat)

# Select stations based on yearly inventory - customize code here
# Input selection criteria for stations (record completeness, etc)
selyrdat <- yrdat[bryrct == 57]   # All States
nrow(yrdat)
nrow(selyrdat)
selstn <- data.frame(StationID = unique(selyrdat$StationID), keep = 'Y')
# rm(selyrdat, yrdat)  # Remove once selections are decided
rm(yrfile)


#**********************************************#
###  CHOOSE FROM THREE DATA FILES TO READ IN  ###
#**********************************************#

# Set Working Directory to access csv data files to plot
setwd(diroutput)

# For Nino Indices, SOI, EQSOI, TNI, and 'BEST', read Year Month data
yrmofile <- paste0('_USCANyrmoinv', SelElem, 'df.csv')
```

```
yrmodat <- read_csv(yrmofile, col_names = TRUE)
yrmodat <- as.data.table(yrmodat)

# For MEI read bimonthly data
MEIfile <- paste0('_USCANbimo', SelElem, 'df.csv')
MEIdat <- read_csv(MEIfile, col_names = TRUE)
MEIdat <- as.data.table(MEIdat)

# For ONI read trimonthly data
ONIfile <- paste0('_USCANtrimo', SelElem, 'df.csv')
ONIdat <- read_csv(ONIfile, col_names = TRUE)
ONIdat <- as.data.table(ONIdat)


#*********************#
#   MONTHLY INDICES    #
#*********************#

# Keep selected stations, merged with selected data set
seldatmo <- as.data.table(left_join(yrmodat, selstn, by = 'StationID'))
nrow(seldatmo)
seldatmo <- seldatmo[keep == 'Y']
seldatmo[ , keep := NULL]
nrow(seldatmo)
seldatmo[, stns := 1]
seldatmo[,base := ifelse(year <= EndBsYr & year >=BegBsYr, 'Y', 'N' )]
seldatmo[,pre82 := ifelse(year < 1982, 'Y', 'N' )]

# Columns to sum by state for Nino, SOI, EQSOI, TNI, BEST
selmocols <- names(seldatmo)[c(14:26)]   #(*Change columns*)
Statedat <- seldatmo[, lapply(.SD, sum, na.rm=TRUE), .SDcols=selmocols,
                     by=list(St, loc, elem, year, month, pre82)]
Statedat[, avgVALm_US := (VALm_US / obs) ]
Statedatmo <- as.data.table(left_join(Statedat, indexmo, by = c('year', 'month')))
IndexName <- names(Statedatmo)[21:32]
IndexName

########## RUN PLOT ################
# Ok to change SelMonth, SelState, and SelIndex at this point.
# Optional Loop - uncomment two lines plus end bracket } to remove
for (indloop in 1:length(IndexName)){
  SelIndex <- indloop

  # Choose the index column to plot
  setnames(Statedatmo, IndexName[SelIndex], 'IndexPlot')

  # Limit data to plot selections
  Statedatplot <- Statedatmo[month == SelMonth & St == SelState & !is.na(IndexPlot)]

  # View range of values for setting y-axis limits
  min(Statedatplot$avgVALm_US)
  max(Statedatplot$avgVALm_US)
  # View range of values for setting x-axis limits
  min(Statedatplot$IndexPlot)
  max(Statedatplot$IndexPlot)

  # Define boundaries of plot
  xlo <- floor(min(Statedatplot$IndexPlot))
  xhi <- ceiling(max(Statedatplot$IndexPlot))
  yhi <- ceiling(max(Statedatplot$avgVALm_US)*100)/100
```

```
  # Select colors and point shapes
  color1 <- 'lightpink4'
  color2 <- 'darkcyan'
  pch1 = 1
  pch2 = 18


  # Labels and Title - revise as needed
  ElemLabel <- subset(ElemLab, ElemLab$Elem == SelElem)
  SelSt <- subset(stabbr, stabbr$St == SelState)
  SelStPlot <- str_to_title(SelSt[2])
  PlotTitle <- paste0(SelStPlot, " ", MoLabel[SelMonth], " ", ElemLabel$ElemName, " vs. ",
              IndexName[SelIndex],"  Index")

  # Widen Plot Region before running plot code
  plot(Statedatplot$IndexPlot, Statedatplot$avgVALm_US, xlab=paste0("Monthly ",
        IndexName[SelIndex], " Index"),
       ylab="average station measurement", xlim=c(xlo, xhi), ylim=c(0, yhi),
       main=PlotTitle,
       pch = ifelse(Statedatplot$pre82=='Y', pch1, pch2), cex.main=1.2, frame.plot=FALSE,
       col=ifelse(Statedatplot$pre82=='Y', color1, color2))
  legend(xlo, yhi, pch=c(pch1, pch2), col=c(color1, color2),
              c("prior to 1982", "1982 and on"),
              bty="o",  box.col="darkgreen", cex=.8)
  # Label outlier points with year - choose boundaries at right and left of plot
  Statedatplot[, outlier := ifelse(IndexPlot > xhi – 0.5 | IndexPlot < xlo + 0.5, year, "")]
  text(Statedatplot$IndexPlot, Statedatplot$avgVALm_US, Statedatplot$outlier, pos=1, cex=0.6)


  # Option: linear regression
  # reg<-lm(avgVALm_US~IndexPlot, data=Statedatplot)
  # abline(reg, lty =2, col = 'grey50')

  # Reset column names in data
  setnames(Statedatmo, 'IndexPlot', IndexName[SelIndex])

}

# In case of error, restore original column names
# colnames(Statedatmo) <- c(names(Statedatmo)[1:20], IndexName)

#**********************#
#         MEI          #
#**********************#

# Keep selected stations, merged with selected data set
seldat <- as.data.table(left_join(MEIdat, selstn, by = 'StationID'))
nrow(seldat)
seldat <- seldat[keep == 'Y']
seldat[ , keep := NULL]
nrow(seldat)
seldat[, stns := 1]
seldat[,base := ifelse(year <= EndBsYr & year >=BegBsYr, 'Y', 'N' )]
seldat[,pre82 := ifelse(year < 1982, 'Y', 'N' )]

# Columns to sum by state for MEI
selcols <- names(seldat)[c(13:25)]   #(*Change columns*)
StateMEI <- seldat[, lapply(.SD, sum, na.rm=TRUE), .SDcols=selcols,
                   by=list(St, loc, elem, year, ord, bimo, MEI, pre82)]
StateMEI <- as.data.table(StateMEI)
StateMEI[, avgVAL2m_US := (VAL2m_US / obs)]
```

```
rm(seldat, MEIdat)

########## RUN PLOT ################
# Ok to change SelBiMo and SelState at this point.
# Limit data to plot selections
# SelState <- 'FL'
# SelMonth <- 2
Month2 <- ifelse(SelMonth == 12, 1, SelMonth+1)
SelBiMo <- paste0(substring(MoLabel[SelMonth], 1, 3), substring(MoLabel[Month2], 1, 3))
StateMEIplot <- StateMEI[St == SelState & bimo == SelBiMo & !is.na(MEI)]
BiMoLab <- paste(substring(SelBiMo, 1, 3), substring(SelBiMo, 4, 6), sep = ' - ')

# View range of values for setting y-axis limits
min(StateMEIplot$avgVAL2m_US)
max(StateMEIplot$avgVAL2m_US)
# View range of values for setting x-axis limits
min(StateMEIplot$MEI)
max(StateMEIplot$MEI)

xlo <- floor(min(StateMEIplot$MEI))
xhi <- ceiling(max(StateMEIplot$MEI))
yhi <- ceiling(max(StateMEIplot$avgVAL2m_US)*100)/100

# Plot MEI graph - be sure to update ranges and title
color1 <- 'lightpink4'
color2 <- 'darkcyan'
pch1 = 1
pch2 = 18

# Labels and Title - revise as needed
ElemLabel <- subset(ElemLab, ElemLab$Elem == SelElem)
SelSt <- subset(stabbr, stabbr$St == SelState)
SelStPlot <- str_to_title(SelSt[2])
MEITitle <- paste0(SelStPlot, " ", BiMoLab, " ", ElemLabel$ElemName, " vs. MEI")


plot(StateMEIplot$MEI, StateMEIplot$avgVAL2m_US, xlab="MEI",
     ylab="average station measurement", xlim=c(xlo, xhi), ylim=c(0, yhi),
     main=MEITitle,
     pch = ifelse(StateMEIplot$pre82=='Y', pch1, pch2), cex.main=1.2, frame.plot=FALSE,
     col=ifelse(StateMEIplot$pre82=='Y', color1, color2))
legend(xlo, yhi, pch=c(pch1, pch2), col=c(color1, color2), c("prior to 1982", "1982 and on"),
       bty="o",  box.col="darkgreen", cex=.8)
# Label outlier points - choose boundaries
StateMEIplot[, outlier := ifelse(MEI > 1.6 | MEI < -1.6, year, "")]
text(StateMEIplot$MEI, StateMEIplot$avgVAL2m_US, StateMEIplot$outlier, pos=1, cex=0.6)

# Option: linear regression
reg<-lm(avgVAL2m_US~MEI, data=StateMEIplot)
abline(reg, lty =2, col = 'grey50')


##### End Program Code


# ========================================================================================== #
# ===== CODE 12 ===================== MAP ENSO INDEX REGIONS ============================== #
# ========================================================================================== #

library(maps)
library(mapproj)  # coordinate grids
```

```
# Pacific-centric Coordinates

# Nino 1+2
CoordPCNino12 <- data.frame(
  lat = c(0, 0, -10, -10, 0),
  lon = c(270, 280, 280, 270, 270)
)

# Nino 3
CoordPCNino3 <- data.frame(
  lat = c(5, 5, -5, -5, 5),
  lon = c(210, 270, 270, 210, 210)
)

# Nino 3.4
CoordPCNino34 <- data.frame(
  lat = c(5, 5, -5, -5, 5),
  lon = c(190, 240, 240, 190, 190)
)

# Nino 4
CoordPCNino4 <- data.frame(
  lat = c(5, 5, -5, -5, 5),
  lon = c(160, 210, 210, 160, 160)
)

# Equatorial SOI - West (5°N-5°S, 220°W-270°W)
CoordPCEQSOI_W <- data.frame(
  lat = c(5, 5, -5, -5, 5),
  lon = c(90, 140, 140, 90, 90)
)

# Equatorial SOI - East   (5°N-5°S, 80°W-130°W)
CoordPCEQSOI_E <- data.frame(
  lat = c(5, 5, -5, -5, 5),
  lon = c(230, 280, 280, 230, 230)
)

########################################################
# NINO INDEX REGIONS - ALL INDICES
###########################################################
# Map Pacific Centric Index SST Region
map("world2", xlim = c(80,300), ylim = c(-40, 40))

# EQSOI Regions
rect(140, -5, 90, 5, col = 'lightcyan1', border = FALSE)
rect(230, -5, 280, 5, col = 'lightcyan1', border = FALSE)

map("world2", xlim = c(80,300), ylim = c(-40, 40), add = TRUE)
map.axes()

map.grid(label = FALSE, lty = 1, col = "grey")
par(ps = 12)
title("El Nino Southern Oscillation Index Regions", family='Times')

par(ps = 10)
lines(x = CoordPCNino12$lon, y = CoordPCNino12$lat, col = "black", lwd = 2)
text(275, -3, "Nino",  family='Times')
text(275, -7, "1+2" ,  family='Times')
```

```
par(ps = 12)
lines(x = CoordPCNino4$lon, y = CoordPCNino4$lat, col = "black", lwd = 2)
text(175, 1, "NINO 4",  family='Times')
lines(x = CoordPCNino3$lon, y = CoordPCNino3$lat, col = "black", lwd = 2)
text(255, 1, "NINO 3", family='Times')
Nino34col <- 'dodgerblue3'
lines(x = CoordPCNino34$lon, y = CoordPCNino34$lat, col = Nino34col, lty = 3, lwd = 3)
text(213, 9, "NINO 3.4 / ONI", col = Nino34col, family='Times', lwd = 3)
#text(213, 9, "NINO 3.4 / ONI / 'BEST'", col = Nino34col, family='Times', lwd = 3)

SOIcol <- 'midnightblue'
DarwinPC <- c( 130.8456, -12.4634)
points(130.8456, -12.4634, cex = 1, col = SOIcol, pch = 19)
par(ps = 8.5)
text(124, -12, "Darwin", family='Times')
Tahiti <- c(210.574, -17.6509)
points(210.574, -17.6509, cex = 1, col = SOIcol, pch = 19)
par(ps = 8.5)
text(205, -15, "Tahiti", family='Times')

par(ps = 12)
lines(x = c(130.8456, 210.574), y = c(-27, -27), col = SOIcol, lwd = 1)
text(170, -30, "SOI", family='Times', lwd = 2)
lines(x = c(130.8456, 130.8456), y = c(-15, -27), col = SOIcol, lwd = 1)
lines(x = c(210.574, 210.574), y = c(-20, -27), col = SOIcol, lwd = 1)

EQSOIcol <- 'cyan4'
TNIcol <- 'mediumpurple4'
par(ps = 12)
text(150, 23, "EQSOI",  family='Times', lwd = 2, col = EQSOIcol)
text(231, 23, "EQSOI",  family='Times', lwd = 2,  col = EQSOIcol)
#text(170, 32, "TNI",  family='Times', lwd = 2, col = TNIcol)
#text(275, -27, "TNI",  family='Times', lwd = 2, col = TNIcol)
par(ps = 9)
text(150, 18, "(Western)",  family='Times', lwd = 1, col = EQSOIcol)
text(231, 18, "(Eastern)",  family='Times', lwd = 1, col = EQSOIcol)
#text(170, 27, "(Western)",  family='Times', lwd = 1, col = TNIcol)
#text(275, -32, "(Eastern)",  family='Times', lwd = 1, col = TNIcol)
arrows(150, 15, 135, 2, length = 0.1, angle = 20, col = EQSOIcol, lwd = 1.8)
arrows(231, 15, 245, -2, length = 0.1, angle = 20, col = EQSOIcol, lwd = 1.8)
#arrows(170, 23, 170, 5, length = 0.1, angle = 20, col = TNIcol, lwd = 1.8)
#arrows(275, -24, 275, -10, length = 0.1, angle = 20, col = TNIcol, lwd = 1.8)



##### End Program Code


# ============================================================================================ #
# ===== CODE 13 ====================== COSTLIEST STORMS =================================== #
# ============================================================================================ #

# Go back to repeat SETUP at top if R has been closed.

# Read in data on Costliest Atlantic Hurricanes
setwd(dirbase)
Costly <- read_csv('CostlyStorms.csv', col_names = FALSE)
Costly <- as.data.table(Costly)
colnames(Costly) <- c("Name", "Cat", "Dmg_USB", "year", "YrNo", "BegDay", "EndDay")
Costly[, Label := paste(year, YrNo, sep = '_')]
# Create columns for formating plot labels
```

```
Costly[, textadj := 0]
Costly[, NameLab := ifelse(Costly$Dmg_USB >=7, Costly$Name, "")]
head(Costly, 10)

head(Costly, 10)
tail(Costly, 10)
#______________________________________________
# Plot All Storms
#______________________________________________
# Sort chronologically
Costly <- Costly[order(year, BegDay),]
Costly$NameLab <- ifelse(Costly$Dmg_USB >=10, Costly$Name, "")
Costly[Name == 'Hugo', 10] <- "Hugo" # Label Hugo
# Adjust text positions and eliminate some names
Costly[Name == 'Hugo', 9] <- - 0.5
Costly[Name == 'Maria', 9] <- 0.5
Costly[Name == 'Charley', 9] <-  -1.8
Costly[Name == 'Wilma', 9] <-  1
Costly[NameLab == 'Matthew', 10] <- ""
Costly[NameLab == 'Rita', 10] <- ""
Costly[NameLab == 'Irma', 10] <-  ""

# Create Bar Plot of all storms (Show in wide plots screen)
j <- barplot(Costly$Dmg_USB, ylim = c(0, 150), col = "darkblue", cex.main = 1.5,
       cex.axis = 0.8, cex.names = 0.7, names.arg = Costly$Label, las = 2,
       ylab = "U.S. Dollars ($ Billions)", xlab = "Year / Storm Number")
j
text(j + Costly$textadj, Costly$Dmg_USB+6, Costly$NameLab, cex = 1.2)
lines(x = c(0, 16), y = c(9.47, 9.47), col = "darkblue", lty = 2, lwd = 1)
text(59, 98, "Irma", cex = 1.2)
arrows(59.5, 93.5, 65.8, 68, col = "black", length = 0.1, angle = 20, lwd = 1.9)

#______________________________________________
# Storms Up to Andrew
#______________________________________________

# Select years and sort chronologically
CostlyA <- Costly[year<=1992,]
CostlyA <- CostlyA[order(year, BegDay),]
head(CostlyA)

# Adjust text positions and eliminate some names
CostlyA[Name == 'Andrew', 9] <-  -1

# Create bar plot of storms (Show in narrow plots screen)
a <- barplot(CostlyA$Dmg_USB, ylim = c(0, 30), col = "darkblue", cex.main = .8,
             cex.axis = .8, cex.names = 0.6, names.arg = CostlyA$Label, las = 2,
             ylab = "U.S. Dollars ($ Billions)", xlab = "Year / Storm Number")
a
text(a + CostlyA$textadj, CostlyA$Dmg_USB+0.9, CostlyA$NameLab, cex = 1)
lines(x = c(0, 16), y = c(9.47, 9.47), col = "darkblue", lty = 2, lwd = 1)


##### End Program Code
```