Course 4 Task 2 Project Instructions

**Task 2: Visualize and Analyze Energy Data**

[INTRODUCTION](https://ut.daacertificate.com/mc/poa?productID=2652&taskID=3335#introduction)

Kathy has asked you to perform an in-depth analysis of the power consumption dataset. You will accomplish this via data visualization and time series regression modeling. As a first step, you will complete a short tutorial on time series analysis. You will then load (if necessary), preprocess, and sample the data. Next you will explore the data using visualization techniques, and you will select the most information-laden visualizations to present eventually to your client. You will then develop three different time series regression models and work with seasonal and non-seasonal forecasting. Finally, you'll summarize your analysis and make recommendations in a final report to your client.

Your deliverable for this task will be a PowerPoint presentation aimed at a business, rather than technical, audience.

============================================================

#### [1. Visualize the data](https://ut.daacertificate.com/mc/poa?productID=2652&taskID=3335" \l "collapsepoa3118)

Before diving into a regression analysis of the energy consumption data, begin your analytical process by using visualization techniques to gain a deeper understanding of the data and to identify possible patterns to pursue via your regression analysis (or analysis using other techniques).

**Work through a tutorial on data visualization**

The [Computerworld tutorial on "painless data visualization](http://www.computerworld.com/article/2497304/business-intelligence-beginner-s-guide-to-r-painless-data-visualization.html)" will quickly get you up to speed on R's built-in visualization functions and also the very popular [ggplot2](http://ggplot2.org/) package.

**Load, convert, and sample the data**

1. Load the [Energy Consumption dataset](http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption)\* into an R dataframe (if you haven't done so already)
2. If needed, preprocess the data using DPLYR and TIDYR or any method you’ve learned thus far in R.
   1. Check the attribute types to see if any need to be converted. (Remember that the data upload process may not always store the attributes as the types you expect.)
   2. Sample the data if appropriate.
      1. The data was captured at one-minute intervals. Do you need data of that granularity to form an accurate picture of electrical usage in a home? If not, what interval is likely to be appropriate for your analysis?
      2. The dataset has 47 months of readings; what questions should you be thinking about?
         1. Should you work with the dataset as a whole, or should you extract one or more shorter time periods?
         2. Think about how you can not only sample the data, but also how you might arrange it
         3. What insight are you trying to gain?
         4. What additional information can you ascertain from the data?
         5. Do you think there will be trends or outliers in the data? Why or why not?
            1. Look at the alternatives presented on the [Subsetting Data page of Quick-R](http://www.statmethods.net/management/subset.html).

Expert Advice



Sampling massive data sets

TIP:

TIP: You can also use chaining and filter() in DPLYR to subset data

**Produce visualizations**

Think about the visualizations that are likely to yield the most insight into the data. You might begin by looking at energy usage by zone during the course of typical days. For example, you might:

1. Pick representative days (e.g., one per season or even one per month) and sample the data to create a new dataframe for each selected day
   1. You can easily use DPLYR to group your data into manageable ‘chunks’ so its easier to manage, manipulate and analyze.
2. (If you haven't already) Pick a reasonable time granularity (e.g., 10 or 15 minutes) and sample the data again
3. Plot each sub-zone's consumption and the overall consumption against time
4. Consider overlaying the plots to see if the overlays are suggestive of the occupant's activities (e.g., the washing is done after dinner is cooked)
5. Compare plots across months and across seasons.

Going further, do an visualization-based analysis by day of the week. Does usage within sub-zones vary by day of the week (e.g., is the laundry done on weekends; are there days that the family typically doesn't cook at home)?

Produce pie chart visualizations that are likely to provide insight, e.g.,

* Percentage of total use at various times of day by each sub-zone.
* Percentage of total power use over a day by each sub-zone.
* Percentage of total power use over an entire year by each sub-zone.

Explore if anything can be learned from bivariate plots of one attribute against another.

Finally, produce any other visualizations that you believe may provide insight.

TIP:

As you do this work, consider the William Cleveland's principles for scientific visualization, the key points for which are:  
1. A graphic should display as much information as possible, with the lowest possible cognitive strain.  
2. Strive for clarity  
    - Avoid too many superimposed elements  
    - Find out the right aspect ratio and scaling  
    - Avoid having the data all skewed to one side or to other side of the graph  
3. Visualization is an iterative process. You will typically try a visualization, and the result will suggest a transformation of the data or another visualization that will produce a more revealing result.

\* This dataset is from the UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

#### [2. Prepare to analyze the data](https://ut.daacertificate.com/mc/poa?productID=2652&taskID=3335" \l "collapsepoa3120)

**Complete a tutorial on time series analysis**

Work through the tutorial in [The Little Book of R for Time Series](https://a-little-book-of-r-for-time-series.readthedocs.org/en/latest/) (also available as a [PDF](https://media.readthedocs.org/pdf/a-little-book-of-r-for-time-series/latest/a-little-book-of-r-for-time-series.pdf)).

**Store your data frame(s) as time series**

Store the data in a time series with appropriate start, end, and frequency. As in the previous step, think about the choosing a sample that makes sense for your ultimate report. Do you need all of the data, or can it be sampled? If so, at what frequency?

TIP:

"Sometimes time series data has been collected at regular intervals that were less than one year, for example, daily, monthly or quarterly. In this case, you can specify the number of times that data was collected per year by using the ‘frequency’ parameter in the [ts()](http://www.statmethods.net/advstats/timeseries.html) function." -- A Little Book of R for Time Series, p. 12.

**Produce time series plots**

Once you have read and stored a time series into R and properly addressed the date and time attributes, the next step is usually to make one or more plots of the time series data, which you can do with the [plot.ts()](https://cran.r-project.org/web/packages/timeSeries/vignettes/timeSeriesPlot.pdf) function in R.

TIP:

The dataset contains some missing values in the measurements. All calendar timestamps are present in the dataset but for some timestamps, the measurement values are missing: a missing value is represented by the absence of value between two consecutive semi-colon attribute separators. For instance, the dataset shows missing values on April 28, 2007. This percentage can be disregarded since it will not statistically affect the general patterns on energy consumption.

Here is a brief video demonstrating plotting a time series in R:

#### [3. Forecasting a time series](https://ut.daacertificate.com/mc/poa?productID=2652&taskID=3335" \l "collapsepoa3123)

https://www.youtube.com/watch?v=YhsNiqvu0Vs

When using regression for prediction, we are often considering time series data with the aim of forecasting the future. A common property of time series data is a trend; using regression we can model and forecast the trend in time series data.

In this step you will create three different time series linear models -- for three different time periods -- using the **tslm** and **forecast** functions found in the [forecast package](https://cran.r-project.org/web/packages/forecast/forecast.pdf) for R, and you will forecast the trends of each time series model you create. (You can find addtional references for the **forecast** and **tslm** packages in the optional resources section as well as additonal resources and examples regarding time series regression.)

TIP:

Before you begin the next step think about the time series you want to represent. If you haven't already, can you remove some of the granularity of the data so it is not so detailed? By doing this you will create a reduced set of data that can be both easily analyzed and easily read when plotting. Since the dataset is so large your graphs may not show enough detail if you do not reduce the sampling of the data before plotting it.

1. Create a time series in R that represents the sample of time you wish to analyze.

TIP:

When choosing time periods to model, think about which are likely to yield the most relevant insight to your business audience.

1. Load the **forecast** package (if you haven't already)
2. Use the **tslm** and **forecast** functions to build a time series linear model and to forecast a time period beyond the dataset.
3. Use the **ggplot2** (or similar) function to visualize the forecast and trend of the linear model
4. Use the **summary** function to record the output of each linear model
5. Repeat for two additonal time series models of two additonal time periods. Experiment with using different confidence intervals and record the variances between all three models listed in the summary for each model.

Your work will produce the following deliverables that will feed into your end-of-task report:

1. One plot showing the forecast and trend for each model (You will submit three plots, in total -- one for each model and trend.)
2. A comparison chart showing the summary outputs for each model.
3. A one-page summary of your understanding of the models and how you interpreted the variances between different confidence intervals.

#### [4. Decomposing a Seasonal Time Series](https://ut.daacertificate.com/mc/poa?productID=2652&taskID=3335" \l "collapsepoa3125)

**Decomposing a Seasonal Time Series**

According to Wikipedia: Seasonal adjustment is a statistical method for removing the [seasonal component](https://en.wikipedia.org/wiki/Seasonality) of a time series that exhibits a seasonal pattern. It is usually done when wanting to analyze the trend of a time series independently of the seasonal components. For example: It is normal to report seasonally adjusted data for unemployment rates to reveal the underlying trends in labor markets. Many economic phenomena have seasonal cycles, such as agricultural production. It is necessary to adjust for this component in order to understand what underlying trends are in the economy and so official statistics are often “seasonally adjusted” to remove seasonal components.

You created three different time series and related forecasts in the previous step, but how can we decompose these time series into useful components?

According to The Little Book of R:

“A seasonal time series consists of a trend component, a seasonal component and an irregular component. Decomposing the time series means separating the time series into these three components: that is, estimating these three components.”

In order to correctly estimate any trend and seasonal components that might be in your time series you need to use the decompose() function in the forecast package, which estimates the trend, seasonal, and irregular components of a time series.

When you use the decompose() function, R returns three different objects that can be accessed from the command line after running decompose() on your time series. For example: If your time series is called ‘timeseriesOne’ the following R objects can be accessed:

* Seasonal component: timeseriesOne$seasonal
* Trend component: timeseriesOne$trend
* Random component: timeseriesOne$random

In this step you will use the three different time series (or whatever datatype you’ve chosen) you previously created (you may also create three new ones, if you’d like) and plot all three components of each time series using the decompose() function. Here is a summary of the steps:

1. Use the time series you previously created or create three new time series in R that represents the sample of time you wish to analyze.

**TIP:** When choosing time periods to model, think about which are likely to yield the most relevant insight to your business audience.

1. Use the **decompose()** function to break each time series into the seasonal, trend and random components
2. Use the **ggplot2** function to visualize the decomposed objects of the time series
3. Use the **summary** function to record the decomposed output of time series.

Your work will produce the following deliverables that will feed into your end-of-task report:

1. Three series of plots, in total – one series for each model. For example: You will have three plots for each time series: one showing the trend, one showing the seasonal component, one showing random or residual component
2. A comparison chart showing the summary outputs for each linear model.
3. A one-page summary of your understanding of the models and how you interpreted the variances between different confidence intervals.

#### [5. Holt-Winters](https://ut.daacertificate.com/mc/poa?productID=2652&taskID=3335" \l "collapsepoa3126)

To make forecasts using simple exponential smoothing in R, you can fit a simple exponential smoothing predictive model using the [HoltWinters() function](https://stat.ethz.ch/R-manual/R-devel/library/stats/html/HoltWinters.html" \t "_blank) in the stats package for R.  The HoltWinters() function returns a list variables, that contains several named elements. (Consult the linked documentation above.)

When you find the non-seasonal patterns, you will create a forecast that predicts the (short-term) energy consumption for the next 30 days following each non-seasonal component.

TIP: You could use seasonal adjustment for this as well since it removes the seasonal component, but you would still need to ensure no seasonality in the trend or residuals

Think of this just as you would when a weatherperson forecasts the weather.

Here is an example of how this would be done procedurally in R, but understand that this example assumes that the data will contain non-seasonal components; it is your job as a data analyst to gain insight about this type of information:

1. Decide how to most logically sample the data for this task; you’ll need four different time series that have no seasonality
2. Use the Holt-Winters Model for exponential smoothing to make forecasts based on each of your newly stored time series.

**TIP:** To use HoltWinters() for simple exponential smoothing, set the parameters beta=FALSE and gamma=FALSE in the HoltWinters() function (the beta and gamma parameters are used for Holt’s exponential smoothing, or Holt-Winters exponential smoothing.

1. Use what you've learned thus far about ggplot2 (or other visualization packages) to create visualizations of your four time series:

For each time series you’ll deliver:

* + One plot for each overall time series analyzed with the Holt-Winters Model.
    - This plot will likely be large and might be hard to interpret; think about how you might sample the data to create a visualization that is easier to interpret and thus more valuable to your audience.
  + One plot that just shows the forecasted portion created from the Holt-Winters Model for each seasonal component. (Think about the weekly-extended weather forecasts; the weatherperson doesn’t usually show you what happened previously in his or her presentation.)
    - You'll use these plots to gain insight for your report.

#### [6. Produce the report to management](https://ut.daacertificate.com/mc/poa?productID=2652&taskID=3335" \l "collapsepoa3127)

You are to create a report that will be presented to a management team and used to drive decisions about energy usage through analysis and forecasting.

At a minimum the presentation should include (these deliverables are aggregated from the previous steps):

* Three or more comparison plots for the variables from the sub-metered household data, illustrating what you believe to be most significant about the consumer's power consumption behavior
  + An explanation of why you believe these to be the most significant
* Voltage Data
* Three decomposed time series plots of the energy usage data that can be used to show patterns and argue for changes in consumer behavior
  + A comparison chart showing the summary outputs for each model.
  + A one-page summary of your understanding of the models and how you interpreted the variances between different confidence intervals
* Four non-seasonal plots showing the Holt-Winters forecast results
* Four non-seasonal plots showing only the forecasted area created by Holt-Winters
* Any useful correlations or predictions you can glean from the data
  + If none can be found or if the relationships are weak, suggest other types of data you believe might be needed to improve the analysis
* Five business recommendations you can suggest based on your visualization and analysis of the power consumption data
  + Include justifications of your recommendations
* One or more slides discussing lessons learned.

**TIP:**

Remember that it is your job as the data analyst not simply to do a bunch of analyses but more importantly to tell management what insight can be gained from the data they have provided to you.