Sophia J Quinton
DSC-540
Dr. Darwiche
Grand Canyon University
3 November 2021

Assignment 1

Gitlab Link - https://github.com/squinton-gcu/Data-Science/tree/main/DSC-540/Assignment1

Part 1 - Tools Readiness

Install Python 3.7 (or later) and PyCharm

Add the following libraries: *Numpy*, *Pandas*, *Matplotlib*, and *Scikit-Learn*

Create simple Jupyter notebooks, in which you import the four packages and write minimal Python scripts demonstrating that all libraries have been installed correctly. You may copy examples from the quickstart tutorials for each one of these libraries:

```
In [2]: #Part1 - Tools Readiness
        ##file from (Larose & Larose, 2019)
        ##pandas
        frame = pd.read_csv("E:/GCU/Graduate Classes/DSC - 540 Machine Learning/Week 1/cereals.cs
        frame.head()
```

Out[2]:

	Name	Manuf	Type	Calories	Protein	Fat	Sodium	Fiber	Carbo	Sugars	...	Weight	Cup
0	100%_Bran	N	C	70	4	1	130	10.0	5.0	6.0	...	1.0	0.3
1	100%_Natural_Bran	Q	C	120	3	5	15	2.0	8.0	8.0	...	1.0	1.0
2	All-Bran	K	C	70	4	1	260	9.0	7.0	5.0	...	1.0	0.3
3	All-Bran_with_Extra_Fiber	K	C	50	4	0	140	14.0	8.0	0.0	...	1.0	0.5
4	Almond_Delight	R	C	110	2	2	200	1.0	14.0	8.0	...	1.0	0.7

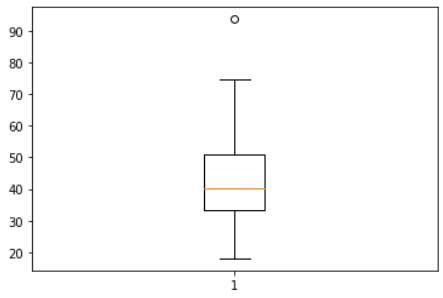
5 rows × 23 columns

```
In [3]: ##numpy
        rounded_rating = np.round(frame['Rating'][0], 3)
        rounded_rating
```

```
Out[3]: 68.403
```

```
In [4]: ##matplotlib
        plt.boxplot(frame['Rating'])
```

```
Out[4]: {'whiskers': [<matplotlib.lines.Line2D at 0x20c3812
          <matplotlib.lines.Line2D at 0x20c3812e8b0>],
         'caps': [<matplotlib.lines.Line2D at 0x20c3812ec40
          <matplotlib.lines.Line2D at 0x20c3812efd0>],
         'boxes': [<matplotlib.lines.Line2D at 0x20c3812e19
         'medians': [<matplotlib.lines.Line2D at 0x20c3813e
         'fliers': [<matplotlib.lines.Line2D at 0x20c3813e7
         'means': []}
```




```
In [5]: ##scikit-learn
        frame_train, frame_test = train_test_split(frame, test_size=0.2, random_state=25)
        frame_train.head()
        print(len(frame_train))
        print(len(frame_test))
```

```
61
16
```

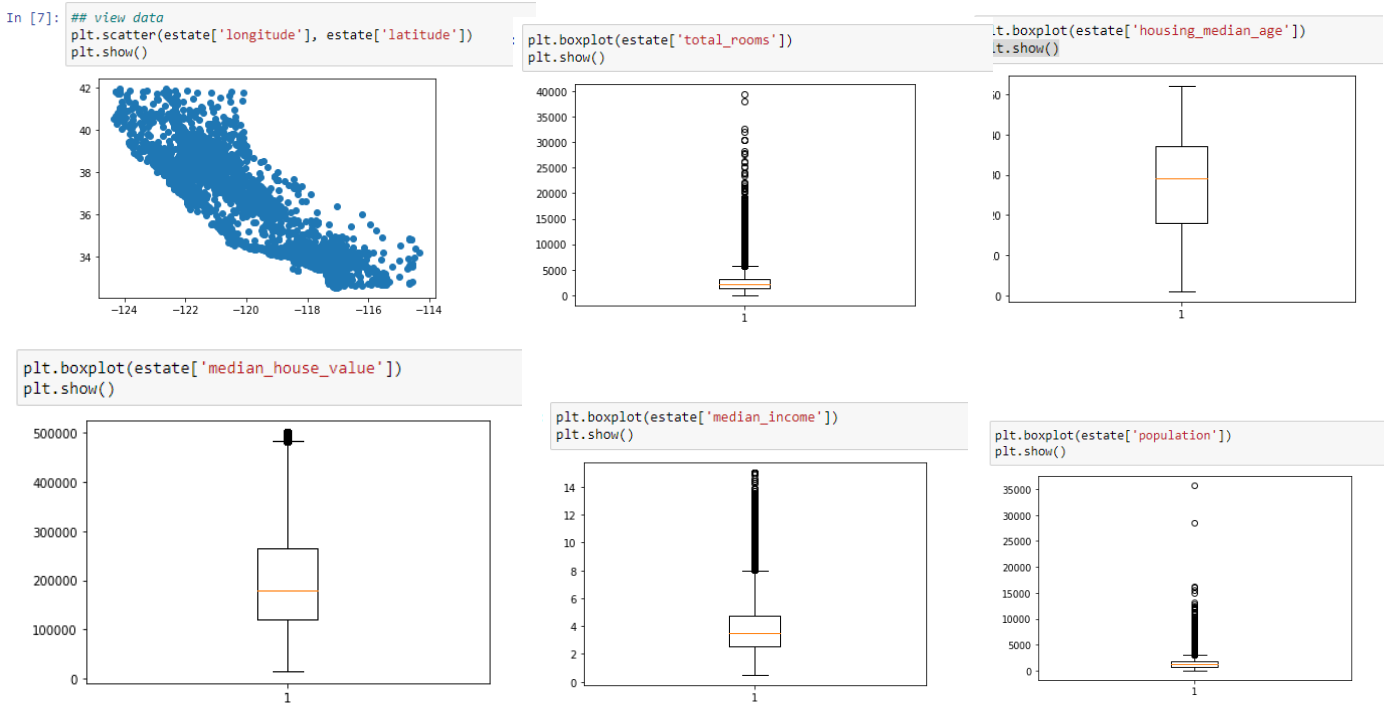
Part 2 - Review Predictive Models and Python Proficiency

Consider the task of calculating the appreciation of a real estate property over time. Using concepts from previous courses (e.g., linear regression, predictive modeling), create a model to predict the future value of the property using at least three different input variables. You have the freedom to decide what these variables are. Show the following:

- **The mathematical model using formal mathematical notation**
- **An explanation of all variables used**
- **An implementation of the model in Python (as a Jupyter notebook), including relevant visual output (e.g., graphs)**

What factors influence the quality of predictions? Discuss these factors and measure the change in outcome when these factors are modified. Estimate the error in your model, both mathematically and in code.

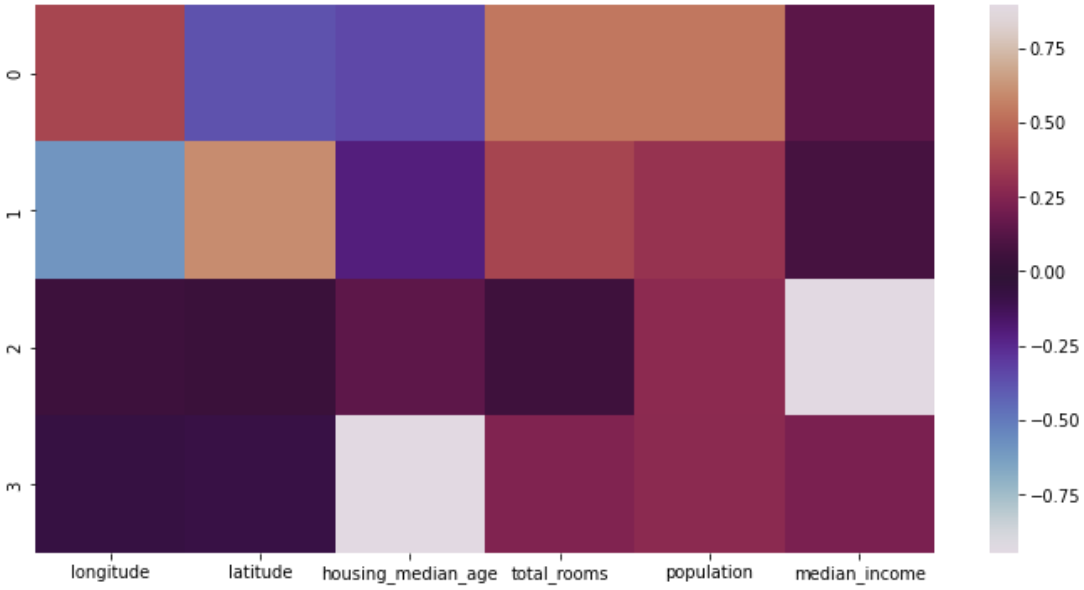
Your computer program should produce a clear quantitative result, error estimate, and a plot that visualizes the prediction.



Before I chose my model, I decided to look at the variables present in the dataset. The longitude and latitude are shown in the scatterplot. The other numerical variables distributions are shown

in the boxplots. Many are skewed in one direction, so a standardization is necessary. There are also differences of the datasets here. The model that I chose was the supervised regression model because the data and the response variable are mostly numeric. This is not the only possible approach to this problem. Another possible model could be a decision tree if the response variable was converted into two separate categories. I believe that a linear regression model is useful here because it does not completely alter the data. The y variable median house value is what is being predicted here. I actually ran the model with all the x values, but there was multicollinearity. I then decided to investigate the multicollinearity (GeeksforGeeks, 2020).

	variable	VIF
0	longitude	592.503040
1	latitude	538.400848
2	housing_median_age	7.238474
3	total_rooms	11.505213
4	population	11.408182
5	median_income	6.264782

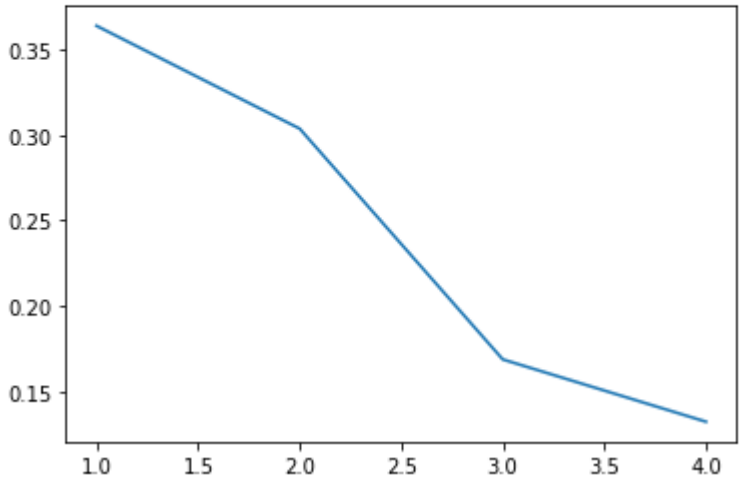


There are extreme versions of multicollinearity in this dataset, so I did a PCA model. The four PCs are represented by the above variables in the heatmap. From the scree plot, the first two components account for most of the variables. So, from this analysis, rooms, population, and latitude help determine the trend of the house market.

Looking at the linear regression of the data, the model is somewhat effective with an adjusted R - squared value of 0.528. This means that the model represents about half of the data points. The coefficients are significant with a low p-value and large t value.

When validating the data with the test set, there is a similarity of the model. The success of the model is determined with the MAE (Mean Absolute Error). If the model MAE is less than the base MAE, then the model is a successful model. The MAE takes into consideration the point of the predictor variable and of the model.

```
: ##scree plot (Zach, 2021)
PC_values = np.arange(pca.n_components_) + 1
plt.plot(PC_values, pca.explained_variance_ratio_)
plt.show()
```




```
Model MAE:  58203.1309674163
base MAE:  78282.93963949986
```

$$Mean\ Absolute\ Error\ = \frac{\sum |y-\hat{y}\}}{n}$$

The error, MSE, is generally the standard deviation of the model.

```
test MSE 78268.50127486532
Model MSE 79554.41649151935
```

$$s = \sqrt{MSE} = \sqrt{\frac{SSE}{n-p-1}} = \sqrt{\frac{\sum(y-\hat{y})\text{^}2}{n-p-1}}$$

From this model, there is a deviation of about $78,000. This is a large error for the model. The model's main predictor variables are the rooms, population and latitude in this dataset. The model is somewhat successful with a R adjusted value of 0.5 and a standard error of 79268. The standard error in the model of each of the components is about 790 for the first component. Again, these errors are large.

```
##run model (Larose & Larose, 2019)
constantX = sm.add_constant(x_estate_train, prepend=True)
estate_model = sm.OLS(y_estate_train, constantX).fit()
estate_model.summary()
```

OLS Regression Results

Dep. Variable:	median_house_value	**R-squared:**	0.528
Model:	OLS	**Adj. R-squared:**	0.528
Method:	Least Squares	**F-statistic:**	4619.
Date:	Mon, 01 Nov 2021	**Prob (F-statistic):**	0.00
Time:	20:11:37	**Log-Likelihood:**	-2.0975e+05
No. Observations:	16512	**AIC:**	4.195e+05
Df Residuals:	16507	**BIC:**	4.196e+05
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.066e+05	619.105	333.762	0.000	2.05e+05	2.08e+05
x1	7677.6593	418.833	18.331	0.000	6856.701	8498.617
x2	934.7505	458.486	2.039	0.041	36.068	1833.433
x3	-7.378e+04	615.501	-119.877	0.000	-7.5e+04	-7.26e+04
x4	4.268e+04	695.343	61.375	0.000	4.13e+04	4.4e+04

Omnibus:	3482.398	**Durbin-Watson:**	1.993
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	8769.875
Skew:	1.158	**Prob(JB):**	0.00
Kurtosis:	5.718	**Cond. No.**	1.66

```
##validate data (Larose & Larose, 2019)
constantX_test = sm.add_constant(x_estate_test, prepend=True)
estate_model_test= sm.OLS(y_estate_test, constantX_test).fit()
estate_model_test.summary()
```

OLS Regression Results

Dep. Variable:	median_house_value	**R-squared:**	0.527
Model:	OLS	**Adj. R-squared:**	0.527
Method:	Least Squares	**F-statistic:**	1149.
Date:	Mon, 01 Nov 2021	**Prob (F-statistic):**	0.00
Time:	17:06:16	**Log-Likelihood:**	-52369.
No. Observations:	4128	**AIC:**	1.047e+05
Df Residuals:	4123	**BIC:**	1.048e+05
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.092e+05	1218.673	171.633	0.000	2.07e+05	2.12e+05
x1	7460.1512	790.853	9.433	0.000	5909.652	9010.650
x2	-205.8746	893.178	-0.230	0.818	-1956.985	1545.235
x3	-7.419e+04	1231.663	-60.240	0.000	-7.66e+04	-7.18e+04
x4	4.184e+04	1351.925	30.947	0.000	3.92e+04	4.45e+04

Omnibus:	809.222	**Durbin-Watson:**	2.045
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	1785.669
Skew:	1.123	**Prob(JB):**	0.00
Kurtosis:	5.310	**Cond. No.**	1.72

Part 3- Technical Report

Refer to the readings in this topic (textbook and the article "The Seven Tools of Causal Inference, with Reflections on Machine Learning," which describe:

- **Four forms of learning (supervised, unsupervised, reinforcement, evolutionary)**
- **Four learning tasks (classification, regression, learning association, clustering)**

- **Seven tools of causal inference**

Examine the article "Using Machine Learning to Translate Applicant Work History into Predictors of Performance and Turnover." Write a two-page technical report covering the following:

1. **Characterize the article in terms of the forms of learning, learning tasks, and causal inference it reports.**
2. **Defend your characterization by mapping the concepts onto the specific details mentioned or inferred in the article.**
3. **Use the GCU digital library to find work describing a form of learning or learning task that is not covered by the categories listed in Part 1. Use your findings to expand your characterization in (2).**

The article, "Using Machine Learning to Translate Applicant Work History into Predictors of Performance and Turnover" by Saijadiani et. al. discusses the use of machine learning to help predict a successful teaching candidate. The article does introduce some controversy when it comes to the overreach of machine learning. The authors collected data from the Minneapolis Public School District between 2007 and 2013 and measured various metrics (Saijadiani et. al., 2019). The first measurement is the work-related experience and classification with ONET, and the next two are related to tenure history and turnover history. The article also uses a few other metrics including performance, student evaluation, expert observations, and standardized test scores (Saijadiani et. al., 2019).

There are four types of forms of machine learning. They include supervised learning, unsupervised learning, reinforced learning, and evolutionary learning. Supervised learning is when prior information is used to create a model that can then be used to predict future data relationships, and it includes classification and regression (Gopal, 2019). Unsupervised learning is often used to help determine similarities with of unknown datasets, which includes clustering and association (Gopal, 2019). Reinforcement learning is when a learning algorithm is modified

or enforced by a good action (Gopal, 2019). An example of this is by making it more difficult to beat a video game because it can anticipate the user's responses. Evolutionary models try to optimize themselves in a variety of ways (Gopal, 2019). In the article by Saijadiani et. al., they use a multitude of supervised learning algorithms for their machine learning method.

> "The naïve Bayes estimator we used achieved a very high correspondence rate of 98.8%, which compares favorable relative to Logistic Regression (98.7% correspondence, a Decision Tree (89.7%), a Random Forest (88.5%), or K-Nearest Neighbor (76.3%)" (Saijadiani et. al., 2019).

Most of the models discussed are considered supervised learning models with one being an unsupervised learning model. The authors in the article provide the models with training data to help determine a predictive model that new data, or test data, can be applied to. The authors state in their article that have data on characteristics including previous jobs, skills, and reasons for leaving a previous job (Saijadiani et. al., 2019). This data can be split into test and train groups that will build each of these models. Then, the model accuracy can be determined. For the unsupervised model, the data is used to understand the categories and similarities.

There are two types of tasks that fall under the category of supervised learning: classification and regression. There is also one example of classification which is used to help with pattern recognition as it groups the response variables based on categorical predictor variables (Gopal, 2019). Regression analysis is used for numeric predictions as it helps determine the output based on fitting the data to a function (Gopal, 2019). The models that the article uses fall into these categories. The naïve Bayes estimator, and the Decision tree are types of classification because they try to recognize patterns in mostly categorical data. Note that decision trees can also categorize with some numeric data, but the response variable is usually

binary. Logistic regression is a type of supervised learning that is used to fit numeric data to a function. The K-Nearest Neighbor model is a type of unsupervised learning that is used to group similar records together (Gopal, 2019).

The casual inference hierarchy is a three-layered pyramid that helps define what questions a model is trying to answer. The three levels of the casual inference hierarchy include the association level (basal), the intervention (middle), and the counterfactuals (top) (Pearl, 2019). Association is focused on looking into the interactions or trends of a relationship (Pearl, 2019). The intervention level is focused on the what ifs of a question, and the counterfactuals look retrospectively to identify causes (Pearl, 2019). There also seven tools that help the success of the models. For the article by Saijadiani et. al., the casual inference that is used is the counterfactuals $P(y_x|x', y,)$. The reason it is a counterfactual model is that it is looking a retrospective data to interpret current or future success. The people that were hired or fired have performance data of the now and also previous job experience. All of this would lead to successful job performance.

Some of the tools that are discussed in Pearl (2019) can also be applied to the article by Saijadiani et. al. One tool that is utilized in this article is the "algorithmization of counterfactuals", but the article is missing or doesn't discuss the other tools (Pearl, 2019). The main issue with this article is that it has a weak demographic bias which eliminates tool 5, and it might not account for confounding variables. This study is limited in that there are many other factors that can determine a successful applicant. Many of these factors include human to human interactions.

A new learning task that isn't discussed in part 1 of this assignment are neural networks which is a subset of machine learning. An article by Liang et. al. (2021) discusses neural

networks as machine learning models. “Currently, neural networks achieve significant success and have wide applications in various machine learning fields like pattern recognition, video analysis, medical diagnosis, and robot control” (Liang et. al., 2021). The authors describe their QNNS, quantum neural networks as being better at parallelism and performance. Neural networks could be applicable to job applicants if the dataset was more robust.

There are some limitations to this model as it has a poor sample pool and confounding variables. This type of article brings to question if a machine learning algorithm is more beneficial than a human process. There might be categories that are left out of the analysis because they are not measurable.

References

DataScience+. (n.d.). *Principal component analysis (PCA) with python*. DataScience+. Retrieved November 2, 2021, from https://datascienceplus.com/principal-component-analysis-pca-with-python/.

Galarnyk, M. (2021, February 3). *PCA using Python (scikit-learn)*. Medium. Retrieved November 1, 2021, from https://towardsdatascience.com/pca-using-python-scikit-learn-e653f8989e60.

GeeksforGeeks. (2020, August 29). *Detecting multicollinearity with VIF - python*. GeeksforGeeks. Retrieved October 31, 2021, from https://www.geeksforgeeks.org/detecting-multicollinearity-with-vif-python/.

Gopal, M. (2019). *Applied machine learning*. McGraw-Hill Education.

Larose, C. D., & Larose, D. T. (2019). *Data science using Python and R*. Wiley.

Liang, Y., Peng, W., Zheng, Z.-J., Silvén, O., & Zhao, G. (2021). A hybrid quantum–classical neural network with deep residual learning. *Neural Networks*, *143*, 133–147. https://doi-org.lopes.idm.oclc.org/10.1016/j.neunet.2021.05.028

Nelson, D. (2021, April 12). *Matplotlib box plot - tutorial and examples*. Stack Abuse. Retrieved October 31, 2021, from https://stackabuse.com/matplotlib-box-plot-tutorial-and-examples/.

Pearl, J. (2019). The Seven Tools of Causal Inference, with Reflections on Machine Learning. *Communications of the ACM*, *62*(3), 54–60. https://doi-org.lopes.idm.oclc.org/10.1145/3241036

Sajjadiani, S., Sojourner, A. J., Kammeyer-Mueller, J. D., & Mykerezi, E. (2019). Using machine learning to translate applicant work history into predictors of performance and

turnover. *Journal of Applied Psychology*, *104*(10), 1207–1225. https://doi-org.lopes.idm.oclc.org/10.1037/apl0000405

Wu, S. (2021, June 5). *What are the best metrics to evaluate your regression model?* Medium. Retrieved November 1, 2021, from https://towardsdatascience.com/what-are-the-best-metrics-to-evaluate-your-regression-model-418ca481755b.

Zach. (2021, September 18). *How to create a scree plot in Python (step-by-step)*. Statology. Retrieved November 2, 2021, from https://www.statology.org/scree-plot-python/.