Analysis of Student Performance

By: Mahmoud Zayad, Parikshit Solunke, Damian Charczuk

Problem Selection

- Our team decided to look at what factors affect the performance of students on subsequent grades.
- The purpose for this project is to provide insight into areas in which schools should focus in on to support their students.

Data Collection

- Data was provided by University of California Irvine pulled from their Machine Learning Repository.
- The data was collected through questionnaires and reports on students' academic performance. It contains thirty-two features and is based on the students performance in the Portuguese language course
- With the data we are trying to predict period three of the students grades.
- There were 650 entries in the dataset.

Data Preparation

- We encountered no issues in regards to the quality of our dataset. No missing values and most features were numeric.
- There were a few categorical features, they were converted to integers by creating dummy variables.

Data Exploration

- Our data had several irrelevant features so we dropped them.
- Those features included: school name, wine/alcohol consumption, reason for going to school, the occupations of the parents, and the relationship the guardians had with the student among other features.
- We determined this by the unique values for certain features and looked at the mean of the final grade (“G3”) column.

```
print(df['famsup'].value_counts())
print(df.groupby(by="famsup")['G3'].quantile([0.25,0.5,0.75]))
```

```
yes    398
no     251
Name: famsup, dtype: int64
famsup
no      0.25    10.0
        0.50    12.0
        0.75    14.0
yes     0.25    10.0
        0.50    12.0
        0.75    14.0
Name: G3, dtype: float64
```

Data Modeling

- We chose to model our data using Regression and Clustering methods.
- For regression we chose to train and test Linear, Ridge and Lasso regression methods.
- For clustering we chose to test various K-means models with varying number of clusters.

Regression Models (I)

- First we built models using all the features in all three methods. We found that Linear and Ridge Regression yielded similar result, however, Ridge regression slightly out performed the Linear model with an adjusted r-squared of ~0.8214.
- The Lasso regression model ignored all the features except “G2” so we decided not to continue with that method.
- We chose to build the rest of the models using Ridge regression from here on out.

Regression Models (II)

- We created various subsets of features to use in the models.
- We intuitively figured that the previous grade periods: “G1” and “G2” would be the greatest predictors for “G3.” Thus we built a model of those first after one using all the features.
- We found that those two features alone had an adjusted r-squared of ~0.874.
- So we included those features in all the models.

Regression Models (III)

- After comparing various subsets we found that the best subset was: G1, G2, studytime, and failures.
- The adjusted r-squared value was ~0.877.
- Other subsets would inc/decrease the r-squared value slightly the lowest being ~0.871 compared to the model of just G1 and G2 with the previously mentioned value of ~0.874.

- This shows that previous performance and study time are the greatest predictors for a students performance.

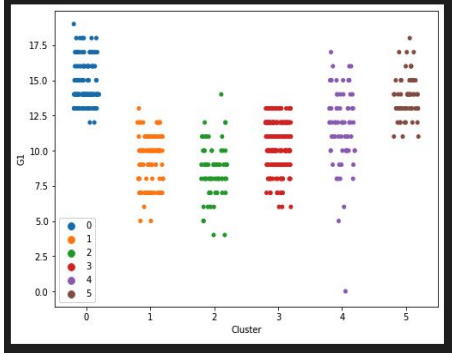
Clustering Models (I)

- We started our clustering process by scaling the features using StandardScaler and choosing G1, G2, address, parent status, and higher as the selected features.
- We ran several K-Means models while incrementing the number of clusters from 3 to 9 to find the number of clusters used for the best performing model.
- Looking at the silhouette scores we determined there are 6 clusters

The figures on the right are the resulting clusters using G1 and G2 Respectively

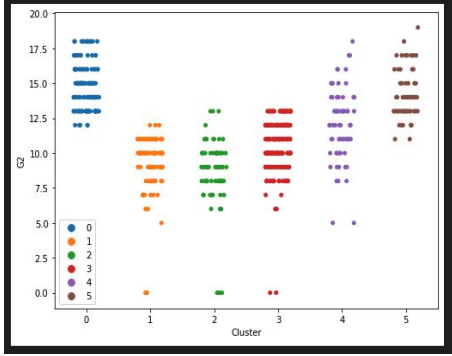
```
#clusters by Grade 1
sns.stripplot('clusters', 'G1', data=df2, hue=clusters,jitter=0.2)
plt.xlabel("Cluster")

Text(0.5, 0, 'Cluster')
```




```
#Viewing clusters by Grade 2
sns.stripplot('clusters', 'G2', data=df2, hue=clusters,jitter=0.2)
plt.xlabel("Cluster")

Text(0.5, 0, 'Cluster')
```



Clustering Models (II)

- We then used AgglomerativeClustering() with the same subset and the hyperparameters of: 6 clusters and using ward linkage

- From the resulting plots, we saw 6 identical clusters, two are relatively high-scoring, two low-scoring, one that is relatively average, and one that has a lot of variation and includes really high scoring students as well as low scoring students.
- We concluded that K-Means and Agglomerative clustering have similar results for our dataset so we decided to move forward using K-Means.

Clustering Models (III)

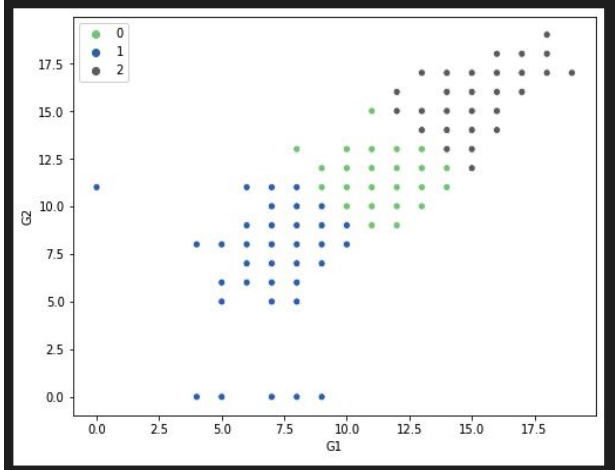
- We ran various other models and determined that there is not a clear pattern for a cluster in terms of the G1 and G2 scores and these are virtually equally distributed across clusters, despite the good silhouette scores.

- So for our last clustering model, we used “G1” and “G2” as our input features.

- The K-Means model using 3 as the number of clusters, k-means++ initialization and 10 iterations results in the following plot, in which we can clearly see the clusters aligning from lower scores for both G1 and G2 to higher scores, with little to no overlap.

```
#View scatterplot for the clusters
sns.scatterplot('G1', 'G2', data=df2, hue=clusters,palette="Accent")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x2521c874ec8>
```



Conclusion

- Through our regression models, we found that the best predictors of the final grade were the earlier grades of the grading period, the amount of time the student spent studying, and the number of past class failures.
- Through our clustering models, we found that G1 and G2 are the best features to use to create accurate clusters.
- No valuable insight was gained from this analysis since it is rather obvious that the grades the student received earlier in the grading period have a large effect on the final term grade. Perhaps if the dataset contained more than a couple hundred data points, we could have received more significant findings.