For the given data set, I have 2 properties and 14 features. Property 1 has numerical values while property 2 type is categorical. Hence, we would be applying two different techniques for the analysis.

For **Property 1**:

1. Linear Regression
2. Multi Linear Regression
3. Principal Component Regression
4. Support Vector Regression

For **Property 2:**
1. Logistic Regression
2. Support Vector Machine

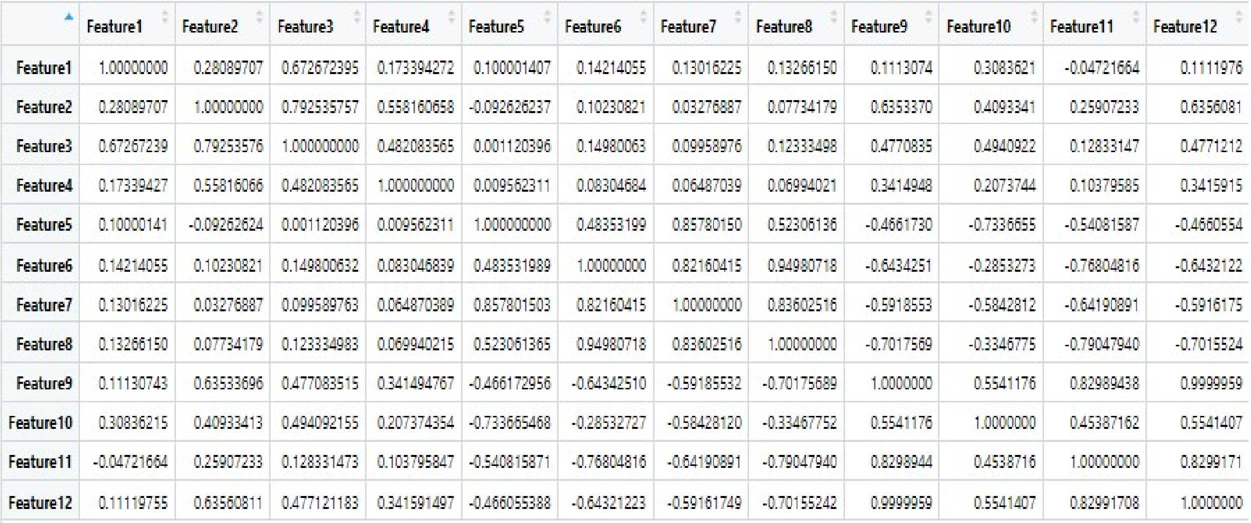
Exploratory Data Analysis:

From the data set provided, I did the following data cleaning,

1. The column Property 2 gives value in categorical terms Yes/No, which needs to be converted in numerical terms. Hence Yes is replaced by 1 and No is replaced by 0.

Yes ------- 1
NO ------- 0

2. We did the **pairwise correlation** and found the following correlation matrix:



▲	Feature1	Feature2	Feature3	Feature4	Feature5	Feature6	Feature7	Feature8	Feature9	Feature10	Feature11	Feature12
Feature1	1.00000000	0.28089707	0.672672395	0.173394272	0.100001407	0.14214055	0.13016225	0.13266150	0.1113074	0.3083621	-0.04721664	0.1111976
Feature2	0.28089707	1.00000000	0.792535757	0.558160658	-0.092626237	0.10230821	0.03276887	0.07734179	0.6353370	0.4093341	0.25907233	0.6356081
Feature3	0.67267239	0.79253576	1.000000000	0.482083565	0.001120396	0.14980063	0.09958976	0.12333498	0.4770835	0.4940922	0.12833147	0.4771212
Feature4	0.17339427	0.55816066	0.482083565	1.000000000	0.009562311	0.08304684	0.06487039	0.06994021	0.3414948	0.2073744	0.10379585	0.3415915
Feature5	0.10000141	-0.09262624	0.001120396	0.009562311	1.000000000	0.48353199	0.85780150	0.52306136	-0.4661730	-0.7336655	-0.54081587	-0.4660554
Feature6	0.14214055	0.10230821	0.149800632	0.083046839	0.483531989	1.00000000	0.82160415	0.94980718	-0.6434251	-0.2853273	-0.76804816	-0.6432122
Feature7	0.13016225	0.03276887	0.099589763	0.064870389	0.857801503	0.82160415	1.00000000	0.83602516	-0.5918553	-0.5842812	-0.64190891	-0.5916175
Feature8	0.13266150	0.07734179	0.123334983	0.069940215	0.523061365	0.94980718	0.83602516	1.00000000	-0.7017569	-0.3346775	-0.79047940	-0.7015524
Feature9	0.11130743	0.63533696	0.477083515	0.341494767	-0.466172956	-0.64342510	-0.59185532	-0.70175689	1.0000000	0.5541176	0.82989438	0.9999959
Feature10	0.30836215	0.40933413	0.494092155	0.207374354	-0.733665468	-0.28532727	-0.58428120	-0.33467752	0.5541176	1.0000000	0.45387162	0.5541407
Feature11	-0.04721664	0.25907233	0.128331473	0.103795847	-0.540815871	-0.76804816	-0.64190891	-0.79047940	0.8298944	0.4538716	1.00000000	0.8299171
Feature12	0.11119755	0.63560811	0.477121183	0.341591497	-0.466055388	-0.64321223	-0.59161749	-0.70155242	0.9999959	0.5541407	0.82991708	1.0000000

- We can observe from the correlation matrix that Feature 5 is correlated with Feature 7 with a correlation value of 0.86,

- Feature 8 is correlated with feature 6 with a correlation value of 0.95 and

- Feature 9 is correlated with feature 12 with a correlation value of 0.99

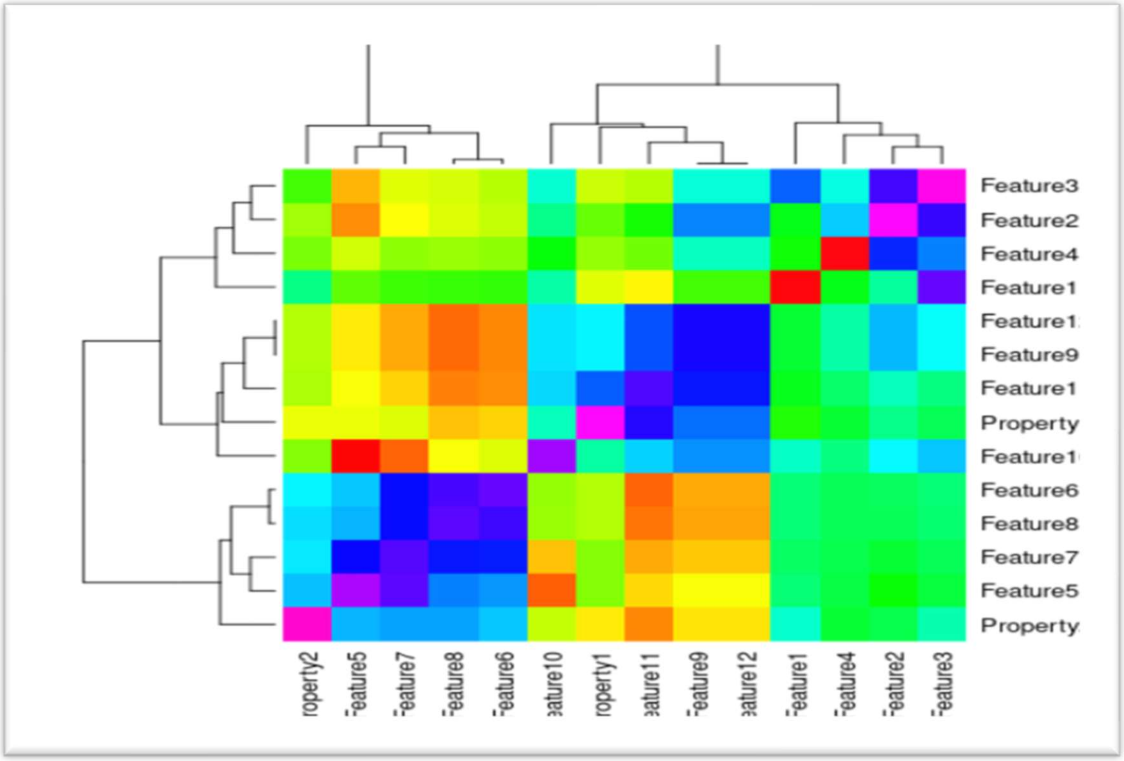
Hence, we can **drop Feature 5, Feature 6** and **Feature 12.**

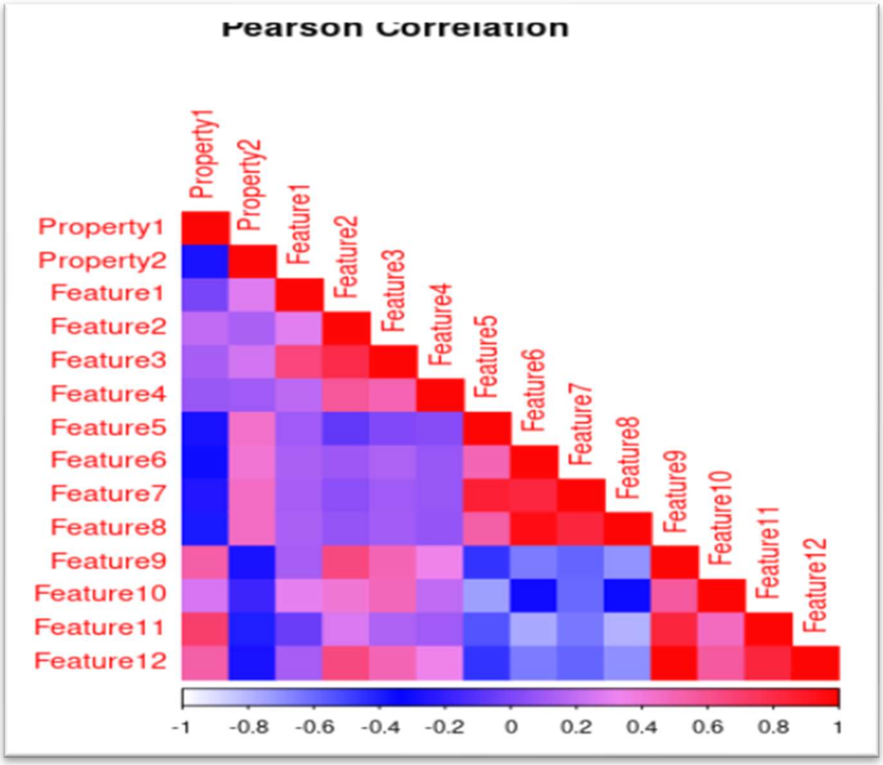
3. **Heatmap**

The below heatmap is plotted using ggplot function in R

Data visualization like below make it easier to comprehend the trends.

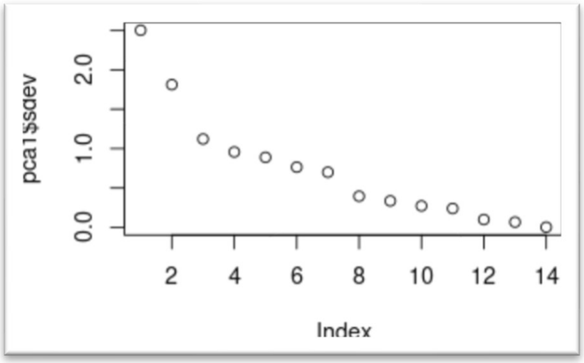
Another importance is the EDA, as it allows us to scan data and understand if there are enough correlations among the features, if the correlation is too unbalanced.



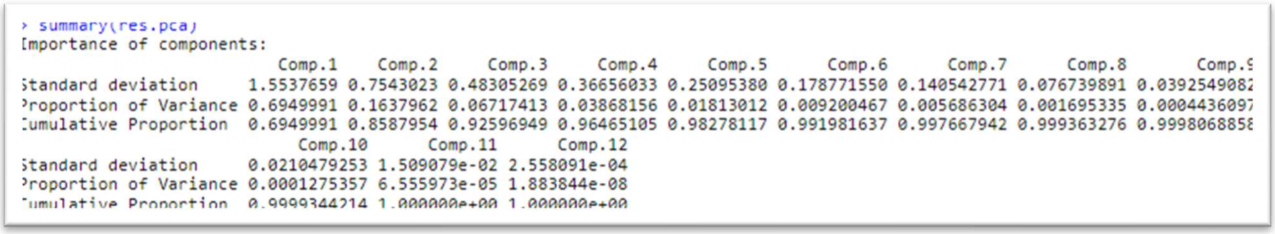


- We can see that Feature 11 has a strong negative correlation with Property1.
- Feature 1 , Feature 2 and Feature 4 are closely clustered together.

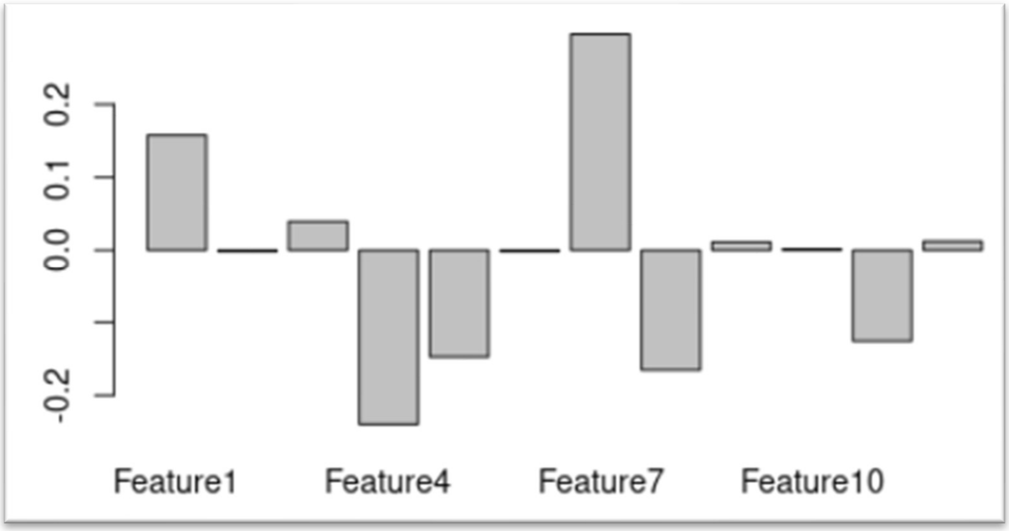
4. **PCA analysis:**



- The plot gives an idea that the first 3 PCs have significantly higher eigen values.
- The elbow starts after PC3.



```
> summary(res.pca)
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4     Comp.5      Comp.6      Comp.7      Comp.8       Comp.9
Standard deviation     1.5537659 0.7543023 0.48305269 0.36656033 0.25095380 0.178771550 0.140542771 0.076739891 0.0392549082
Proportion of Variance 0.6949991 0.1637962 0.06717413 0.03868156 0.01813012 0.009200467 0.005686304 0.001695335 0.0004436097
Cumulative Proportion  0.6949991 0.8587954 0.92596949 0.96465105 0.98278117 0.991981637 0.997667942 0.999363276 0.9998068858
                            Comp.10      Comp.11      Comp.12
Standard deviation     0.0210479253 1.509079e-02 2.558091e-04
Proportion of Variance 0.0001275357 6.555973e-05 1.883844e-08
Cumulative Proportion  0.9999344214 1.000000e+00 1.000000e+00
```

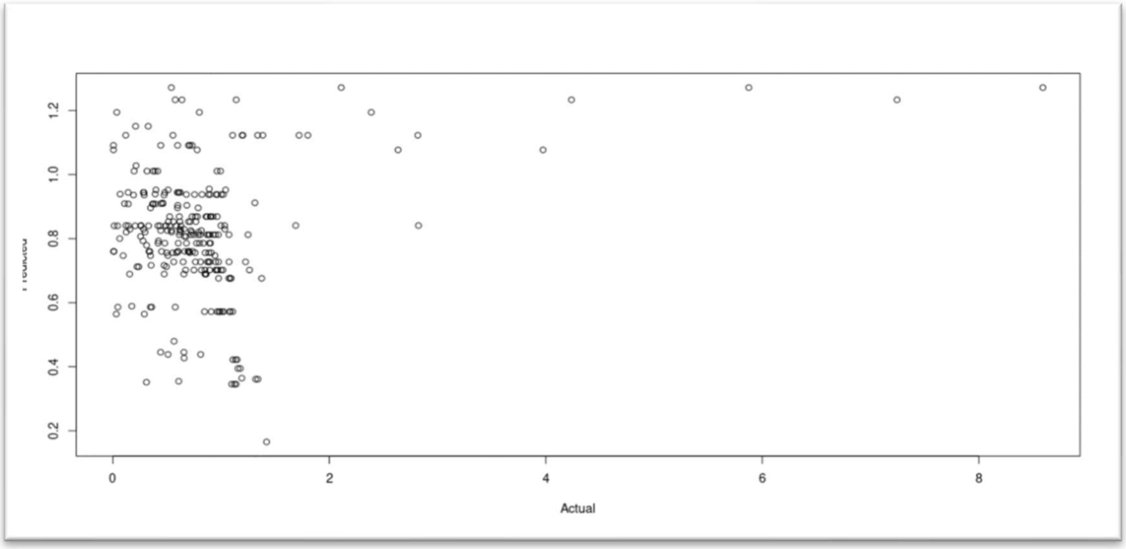


- We can observe that Feature 4 has a strong negative correlation.
- Feature 1 has the highest importance based on the bar chart.
- Hence, we can keep Feature 4, Feature 7, Feature 1, Feature 6 and Feature 10 as they are the most influential features.

Property 1

- For all the regression models the data is divided into 2 test data and training data.
- The training data consists of 80% of the overall data.
- The first 10 rows data of features have been excluded from the analysis.

1. **Linear Regression:**
   Based on the common Features from pairwise correlations and PCA analysis. I preferred to go with Feature 1 as it had the highest importance in the PCA analysis and has less correlation with the property which suggests a great variation in the data.

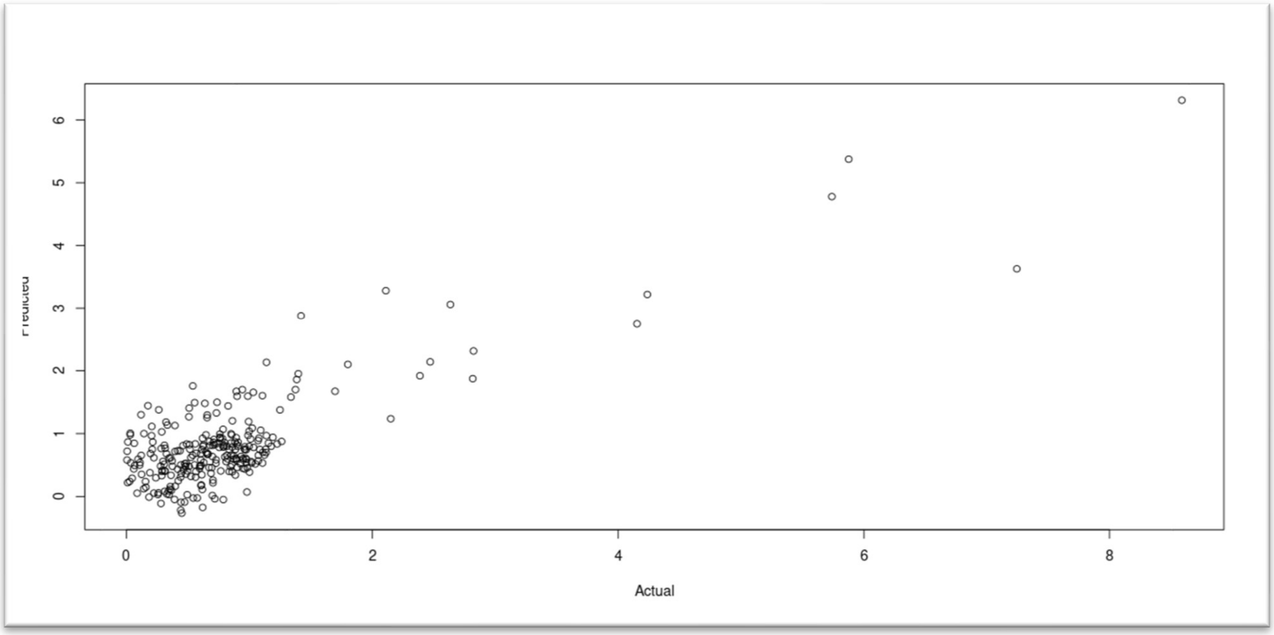


Linear Regression Model	Values
RMSE TRAIN	84.66%
RMSE TEST	82.46%
R^2 Value	0.05%

- Based on the RMSE values and the R-squared values, it signifies that the model might not be able to fit the data very well. We would need to explore other features or will have to add some features to get a highly accurate model.

- Most of the predicted values are clustered around the range of 0 – 2, when the actual values are closer to 0. This signifies that the model tends to over-predict the data.

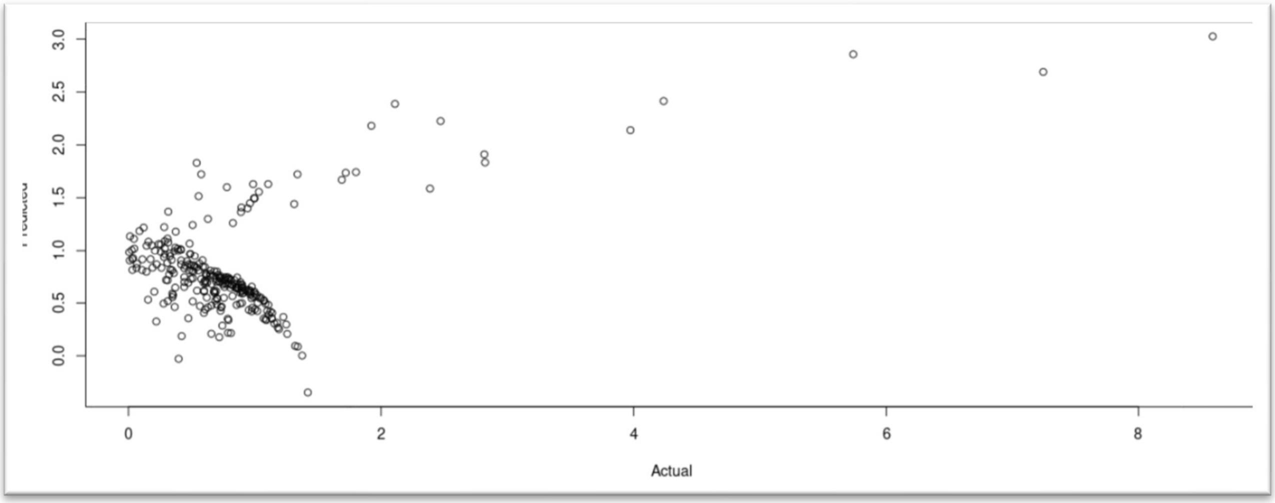
<u>2. Multi Linear Regression:</u>



Multi Linear Regression Model	Values
RMSE TRAIN	42.21%
RMSE TEST	41.41%
R^2 Value	68.11%

- The model provides a reasonably good fit for data. It captures more than two thirds of the variance.
- Based on the R^2 Value, all the features do have a significant influence on the property.

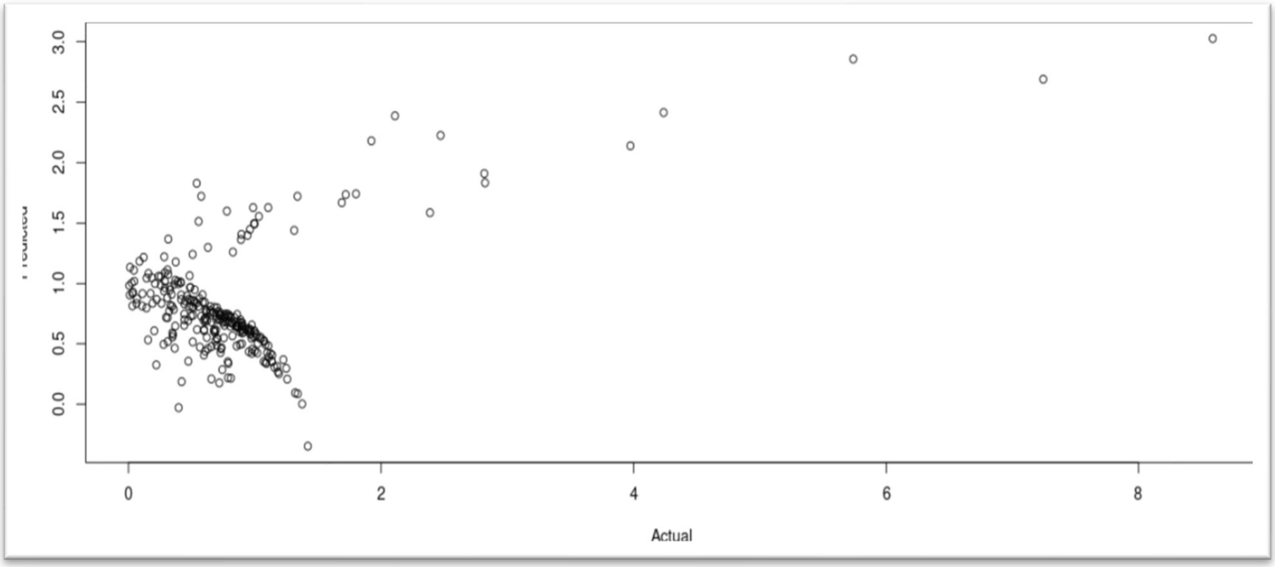
3. Principal Component Regression:



Principal Component Regression Model	Values
RMSE TRAIN	73.28%
RMSE TEST	73.08%
R^2 Value	28.47%

- The PCR model incorporates all features reduced into principal components.
- The model demonstrates consistency with high value of error for both the training and test data sets.

4. Support Vector Regression:



Support Vector Regression Model	Values
RMSE TRAIN	29.67%
RMSE TEST	27.21%
R^2 Value	88.27%

- It is quite evident that there is a strong correlation between the actual and predicted values.
- The average discrepancies between training data set and test data are minimal.
- Overall, the model provides robust performance in predicting the value of property 1.

The following table represents the comparison of each model in terms of accuracy with trained and unseen test data set.

	Linear Regression	Multi Linear Regression	Support Vector Regression	Principal Component Regression
RMSE TRAIN	84.66%	42.21%	29.67%	73.28%
RMSE TEST	82.46%	41.41%	27.21%	73.08%
R^2 Value	0.05%	68.11%	88.27%	28.47%

Indication to how generalizable each model is, and how sensitive it is to the parameters used in the regression model,

Linear Regression:

- The generalizability is low due to a very high RMSE value.
- Sensitivity is too high around feature 1.
- Since the model uses just one feature, there is room to understand the sensitivity with additional features.

Mutli Linear Regression:

- The model shows moderate generalizability. The models show a great variance. With less difference in RMSE value on test and training data, the model shows a great generalization.
- The model's sensitivity is moderate as it uses multiple features.
- The model is reasonably stable due to less difference in RMSE value over test and training data set.

Principal Component Regression:

- Low generalizability due to low R-squared value as majority of variance remains unexplored.
- Low sensitivity as PCR uses principal components in the analysis.

Support Vector Regression:

- High generalizability as the model has the highest R-squared value and lowest RMSE value for both training and test data set.

- Less difference in the RMSE value suggests that model is less sensitive to the to variations in the parameters.

I will **plan to deploy the SVR model** for the following reasons:

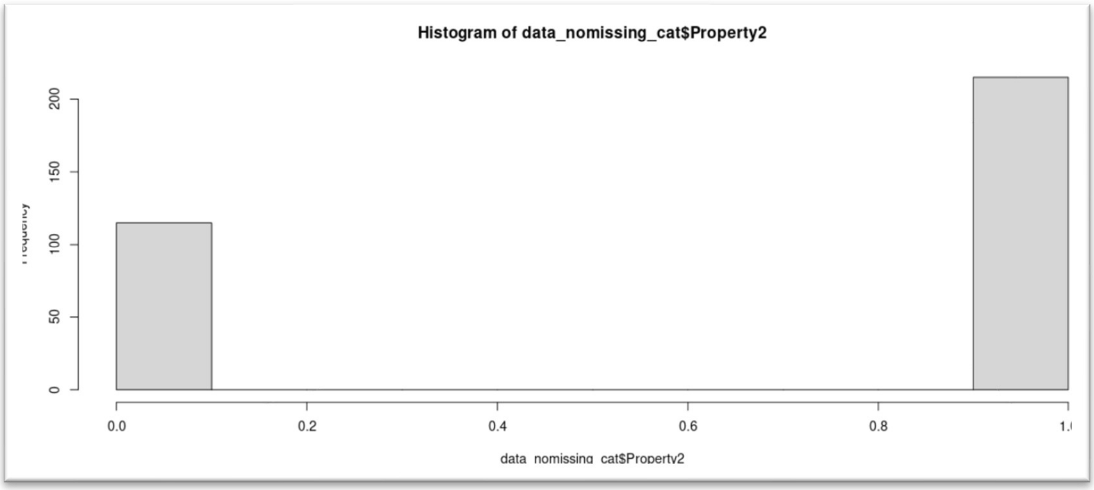
- High R-squared value: The impressive value of 88.27% demonstrates that it captures significant portion of the variance in the data.

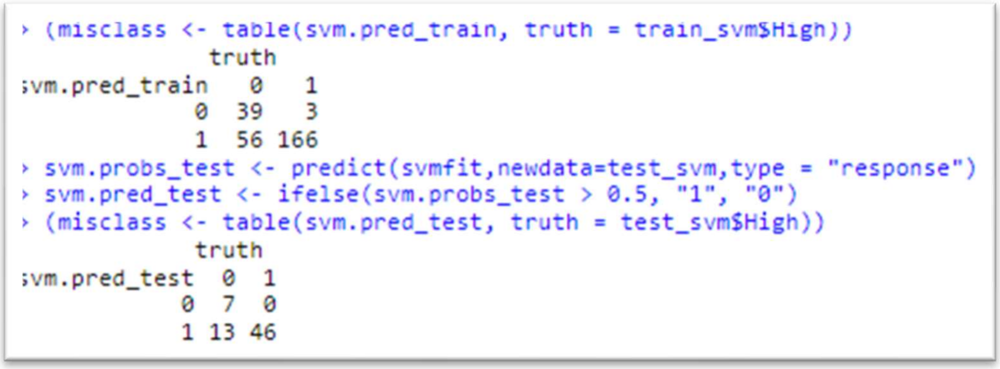
- The model has lowest RMSE values for both training and test data sets, which signifies better fit and prediction of the data.

- Overall, I am quite confident that model will be a perfect fit apart from the computational cost that the model might occur.

Categorical Regression for <u>Property 2</u>:

1. **<u>Support Vector Machine:</u>**





```
> (misclass <- table(svm.pred_train, truth = train_svm$High))
              truth
svm.pred_train   0   1
             0  39   3
             1  56 166
> svm.probs_test <- predict(svmfit,newdata=test_svm,type = "response")
> svm.pred_test <- ifelse(svm.probs_test > 0.5, "1", "0")
> (misclass <- table(svm.pred_test, truth = test_svm$High))
             truth
svm.pred_test  0  1
            0  7  0
            1 13 46
```

The model data demonstrates-

For training data set:

Predicted as class 0 (Negative class):
TN (True Negatives): 39 instances were correctly predicted as class zero.
FN (False Negatives): 3 instances were incorrectly predicted as class zero when they belonged to class 1.

Predicted as class 1 (Positive class):
FP (False Positives): 56 instances were incorrectly predicted as class 1 when they belonged to class 0.
TP (True Positives): 166 instances were correctly predicted as class 1.

For test data set:

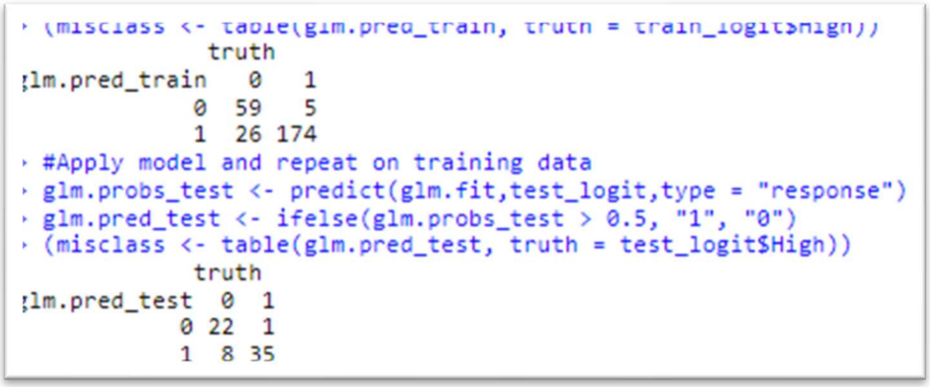
Predicted as class 0 (Negative class):
TN (True Negatives): 7 instances were correctly predicted as class zero.
FN (False Negatives): no instances were incorrectly predicted as class zero when they belonged to class 1.

Predicted as class 1 (Positive class):
FP (False Positives): 13 instances were incorrectly predicted as class 1 when they belonged to class 0.
TP (True Positives): 46 instances were correctly predicted as class 1.

Conclusion:

The model exhibits high sensitivity for the positive class.
The model is biased towards class 1.
Excellent performance ability to detect the positive class on the test data.

2. Logistic Regression



```
> (misclass <- table(glm.pred_train, truth = train_logit$High))
              truth
glm.pred_train   0   1
             0  59   5
             1  26 174
> #Apply model and repeat on training data
> glm.probs_test <- predict(glm.fit,test_logit,type = "response")
> glm.pred_test <- ifelse(glm.probs_test > 0.5, "1", "0")
> (misclass <- table(glm.pred_test, truth = test_logit$High))
             truth
glm.pred_test  0  1
            0 22  1
            1  8 35
```

The model data demonstrates-

For training data set:

Predicted as class 0 (Negative class):
TN (True Negatives): 59 instances were correctly predicted as class zero.
FN (False Negatives): 5 instances were incorrectly predicted as class zero when they belonged to class 1.

Predicted as class 1 (Positive class):
FP (False Positives): 26 instances were incorrectly predicted as class 1 when they belonged to class 0.
TP (True Positives): 174 instances were correctly predicted as class 1.

For test data set:

Predicted as class 0 (Negative class):
TN (True Negatives): 22 instances were correctly predicted as class zero.
FN (False Negatives): 1 instance was incorrectly predicted as class zero when they belonged to class 1.

Predicted as class 1 (Positive class):
FP (False Positives): 8 instances were incorrectly predicted as class 1 when they belonged to class 0.
TP (True Positives): 35 instances were correctly predicted as class 1.

Conclusion:

- The model exhibits strong sensitivity for the positive class.
- The logistic model is more balanced as compared to SVM.
- Strong predicting balance between positive and negative classes.

Based on the conclusion discussed, I would go with Logistic regression method. As it outperforms SVM in the given case.

- The model represents fewer misclassifications due to more balanced predictions.

- Strong sensitivity and minimized false negatives and positives on the test data indicates its reliable performance and potential for generalizing well on an unseen data set.

Summary:

Best model for

Property 1 – SVR (Support Vector Regression)
Property 2 – Logistic Regression

Hence the predicted values of the first 10 rows are:

Property 1	6.24	2.15	2.47	1.92	1.37	5.87	2.81	4.15	3.97	5.73
Property 2	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes