Large Scale Data Mining: Models and Algorithms: Project #4

Due on March 6th, 2020 at 11:59pm

Professor Roychowdhury, Vwani

Wang, Yin, He, Kang

Question 1

In this project, we will explore common practices for best performance of regression. We will conduct different experiments and identify the significance of practices that we will discuss in the following sections. In this project, we will use two different datasets. The first one is a bike sharing dataset. Further description of this dataset can be found in this link: https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset. In this step, we will do some analysis on three different labels:

1. casual: count of casual users.
2. registered: count of registered users
3. cnt: count of total rental bikes including both casual and registered

We will perform data inspection for all three targets and continue the project with the target 'cnt'. The second dataset is a video transcoding time dataset, which basically includes input and output video characteristics along with their time taken for different valid transcoding. The detailed description of each features in this dataset can be found in the project handout, thus we will not discuss them in this section. The target variable for this dataset is 'utime' which is the total transcoding time for transcoding.

In this question, we will plot a heatmap of Pearson correlation matrix of each dataset's columns, and report the feature that have the highest absolute correlation with the target variable. The two heatmaps are shown below as Figure 1, 2.

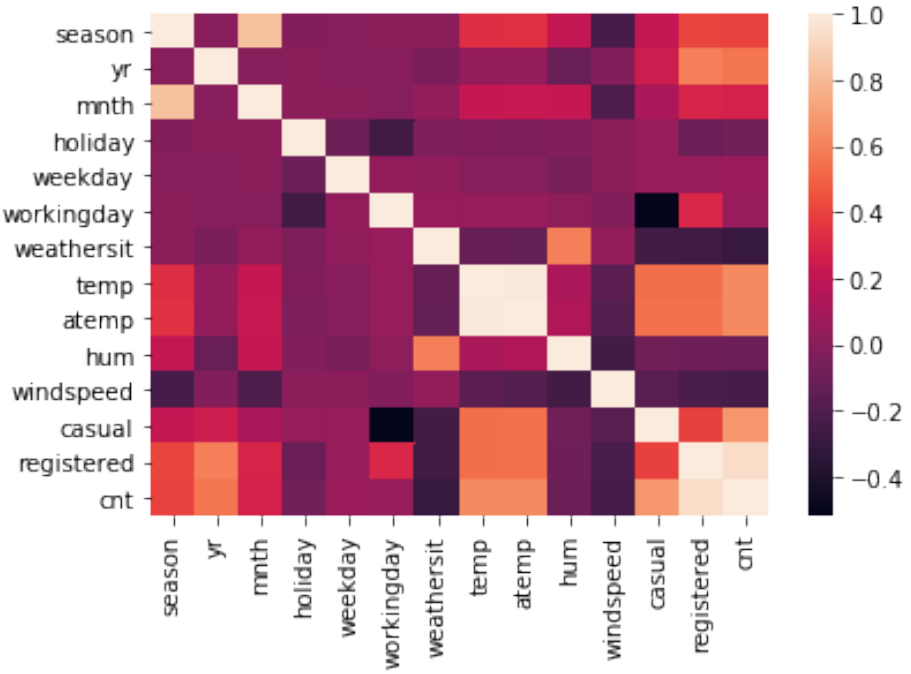

Figure 1: The heatmap of Pearson correlation matrix of bike data set.

For the bike dataset, by inspecting the correlation matrix, which can be found in our notebook file, we found out that for the target 'casual', feature 'cnt' has the highest absolute correlation; for the target 'registered', feature 'cnt' has the highest absolute correlation; for the target 'cnt', feature 'registered' has the highest absolute correlation, which make sense because we know that feature 'cnt' is the sum of 'casual' and

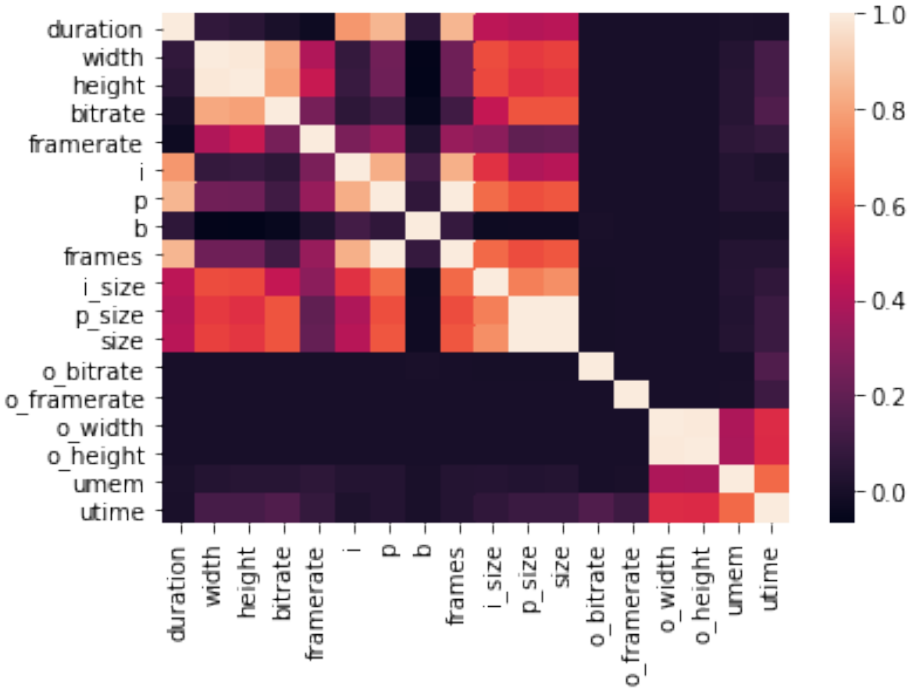

Figure 2: The heatmap of Pearson correlation matrix of video data set.

'registered', thus there is no doubt that this will happen. On the other hand, if we only consider feature 'cnt' as our target variable and exclude 'casual' and 'registered' in our matrix, then feature 'atemp' has the highest absolute correlation, which means, by the definition of this feature, normalized feeling temperature in Celsius is strongly associated with the total number of rental bikes. For the video dataset, by inspection the correlation matrix, which can be found in our notebook file, we found out that see that feature 'umem' has the highest absolute correlation, which implies that total codec allocated memory for transcoding is strongly associated with total transcoding time for transcoding.

Question 2

In this question, we will plot the histogram of numerical features. For the bike dataset, by inspecting the description of each feature, we found out that 'season','yr','mnth','holiday','weekday','workingday',and 'weathersit' are categorical features, thus excluding them in this question. The histograms are shown as Figure 3.

For the video dataset, by inspecting the description of each feature, we found out that 'codec' and 'o_codec' are categorical features, thus excluding them in this question. The histograms are plotted as Figure 4.

To answer the question of handling high skewness, we can first separate skewness into **positively skewed** and **negatively skewed**. Positively skewed is when the tail of the distribution is on the right, on the other hand negatively skewed is when the tail of the distribution is on the left. If the data is positively skewed, we can perform cube root, square root, or logarithm transformations. Note that in logarithm transformation, we can use x to log base 10 of x, or x to log base e of x (lnx), or x to log base 2 of x. If the data is negatively skewed, we can perform square, cube root or logarithm transformations. Note that in general, to resolve the skewness problem, we can also try to identify the outliers and possibly remove them.

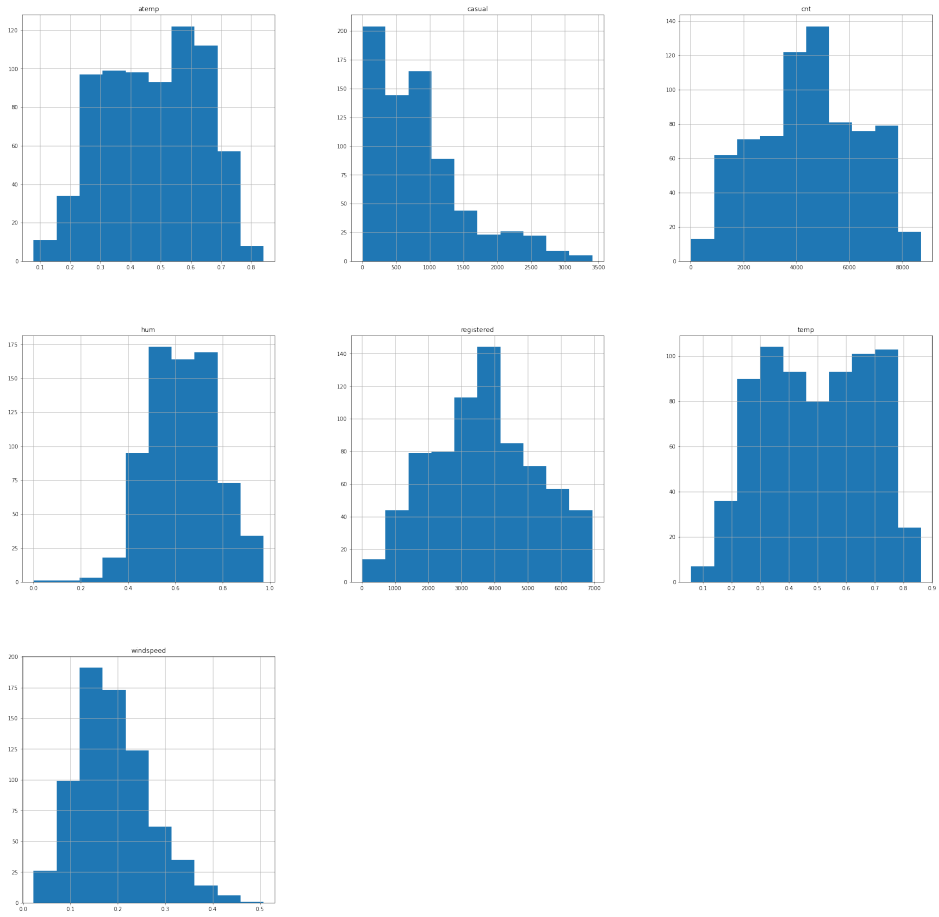

Figure 3: The histogram of numerical features for bike data set.

Question 3

In this question, we will plot the boxplots of categorical features vs target variable for each dataset. For the bike dataset, categorical features are 'season', 'yr', 'mnth', 'holiday', 'weekday', 'workingday', 'weathersit' while the targets are 'casual', 'registered' and 'cnt'. The corresponding box plot for different targets are shown as Figure 5, 6, 7.

For the video dataset, categorical features are 'codec' and 'o_codec', and the plots are shown as Figure 8.

We can see from the plots that for the bike dataset, when we only considering 'cnt' as our target variable, there is almost no outliers in the box-plots, however looking at the box-plots for the video dataset, there are many outliers for 'codec' and 'o_codec', implying that the distributions are skewed.

Question 4

In this question, we will plot the count number per day in bike dataset for a few months. In our case, in order to identify any repeating patterns in every month, we plotted from January to May. The plots are shown as Figure 9, 10, 11, 12 and 13.

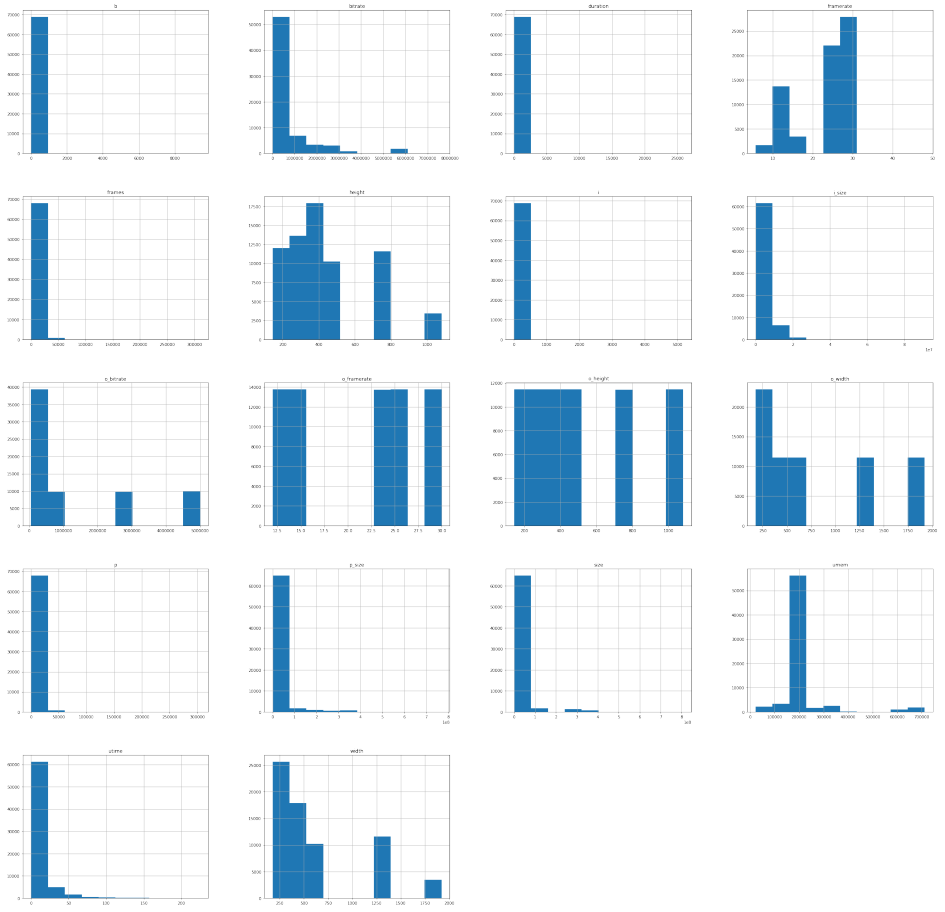

Figure 4: The histogram of numerical features for video data set.

Question 5

In this question, we will plot the distribution of video transcoding times in the video dataset. The plot is shown as Figure 14.

From the above plots, we can see that compare to the begin and the end of each month, there is a peak around the time period after the middle of each month and before the end of each month. Moreover, it seems that the peak value always achieves at weekend or holiday.

From the above plot, we can see that almost all the data in 'utime' lie between $0 - 50$, and we can also see that most of the data lie around $0 - 5$. This shape of distribution can be confirmed by noticing that the mean of the transcoding times is 9.996, and the median transcoding times is 4.408.

Question 6

In order to train the dataset, sometimes we need to handle the categorical features. Common pre-processing step is to convert categorical features into numbers. In general, there are two ways to convert categorical features into numbers, the first method is to assign a scalar, for example if the feature is "Quality" related, then we can assign numbers from 1 to 5 to represent quality from 'Poor' to 'Excellent'. On the other hand, if there is no numerical meaning behind the categorical feature, then we can use 'one-hot encoding'. As

Question 6 continued on next page...

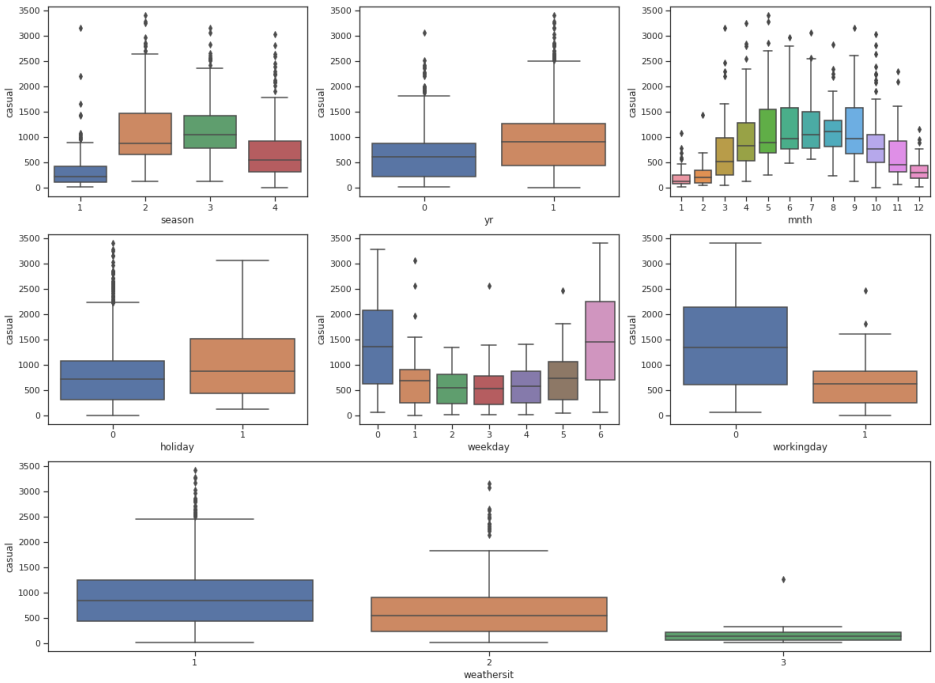

Figure 5: The box plot of categorical vs casual rentals.

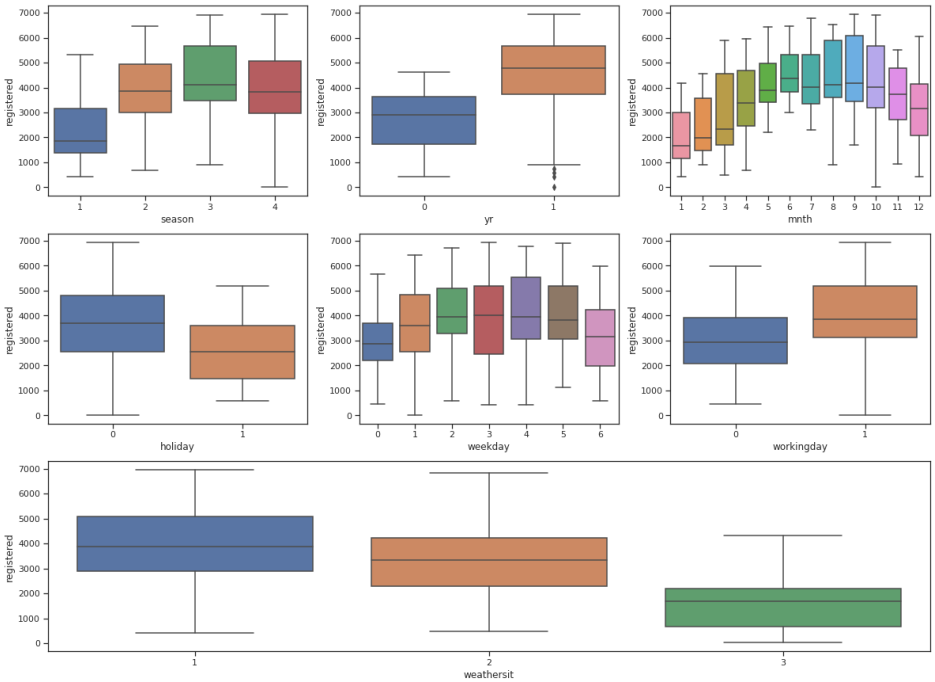

Figure 6: The box plot of categorical vs registered users' rentals.

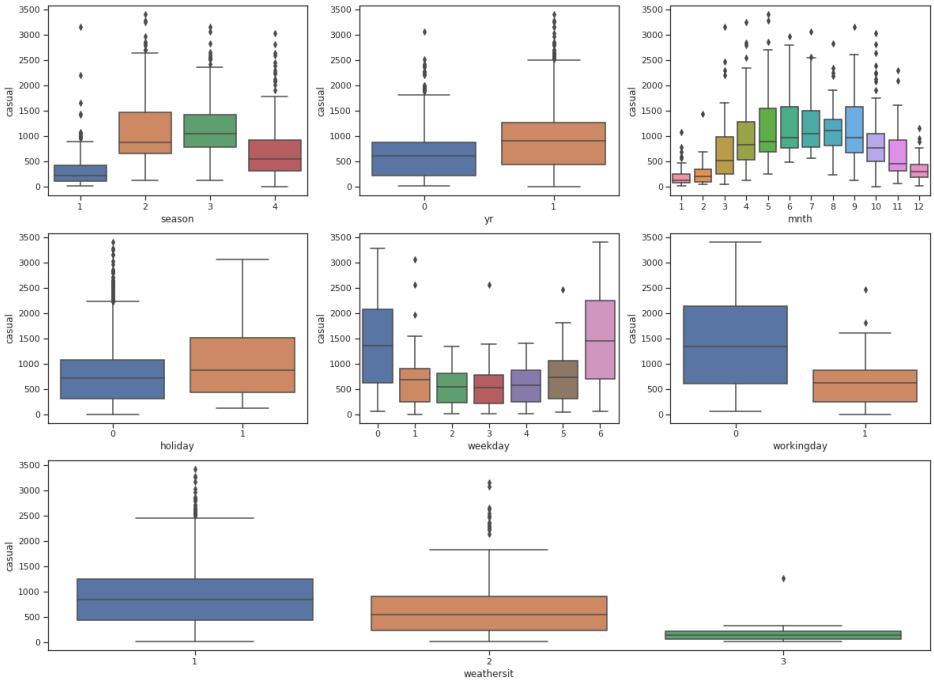

Figure 7: The box plot of categorical vs total count of rentals.

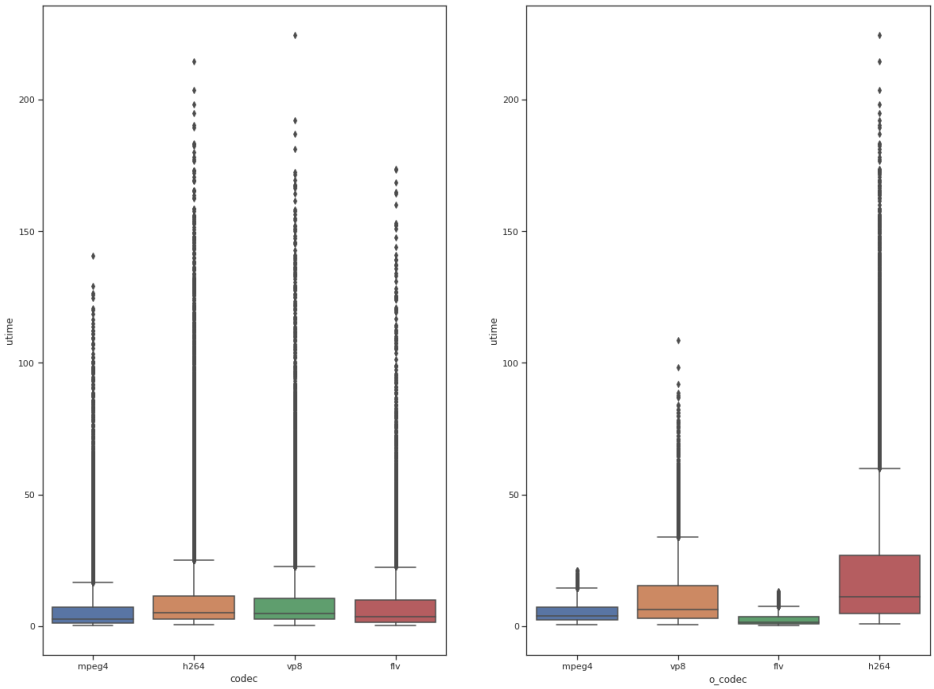

Figure 8: The box plot of categorical vs transcoding time.

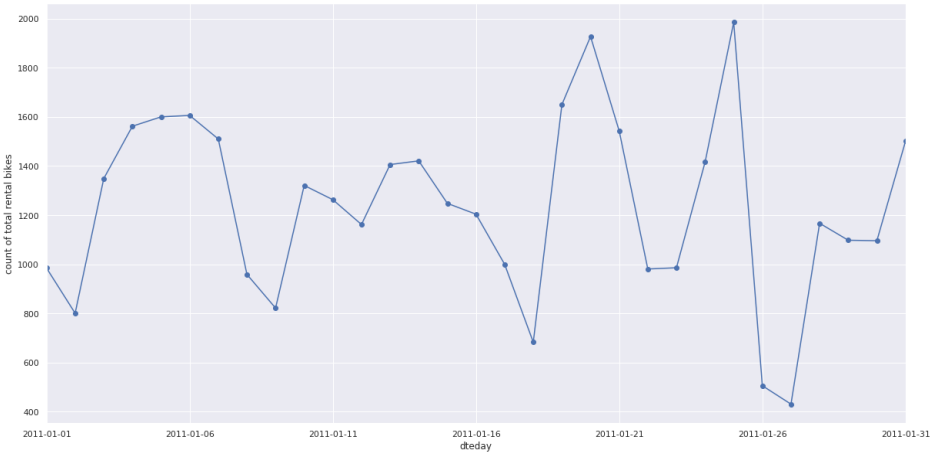

Figure 9: The count number per day in bike dataset for January.

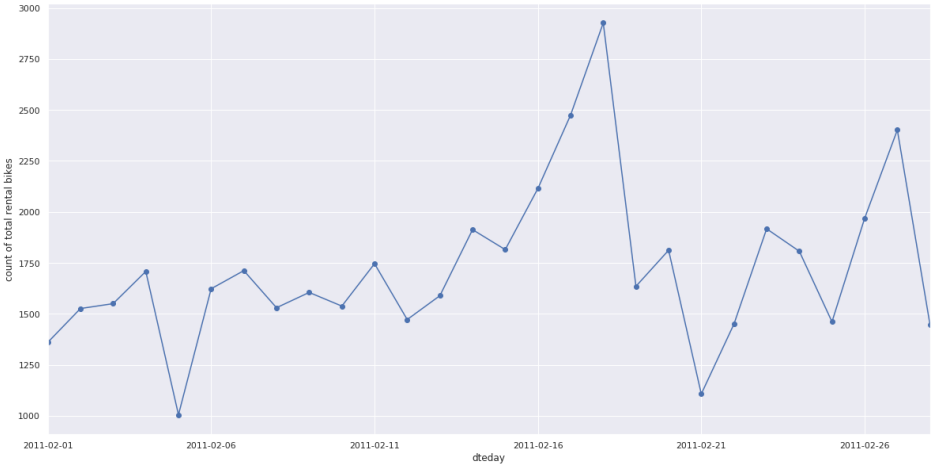

Figure 10: The count number per day in bike dataset for February.

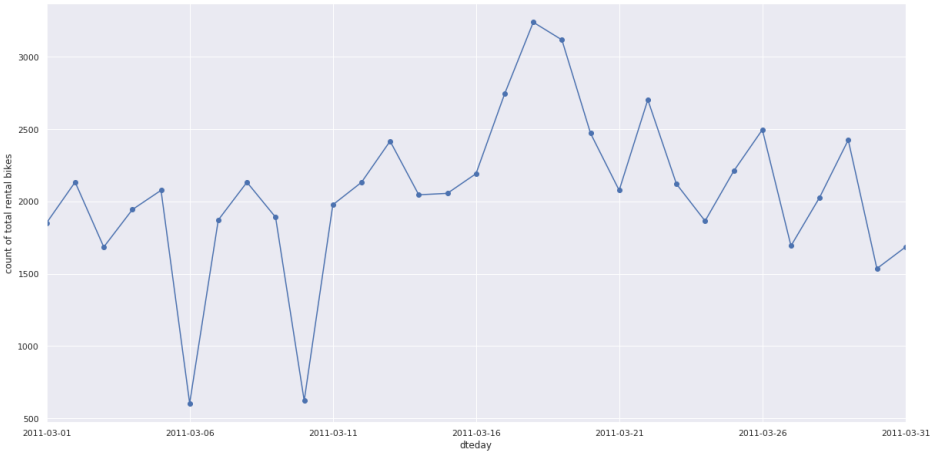

Figure 11: The count number per day in bike dataset for March.

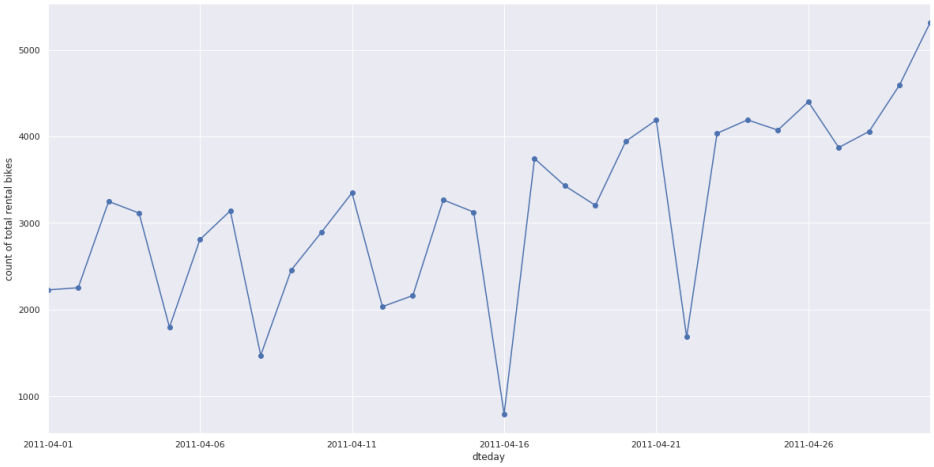

Figure 12: The count number per day in bike dataset for April.

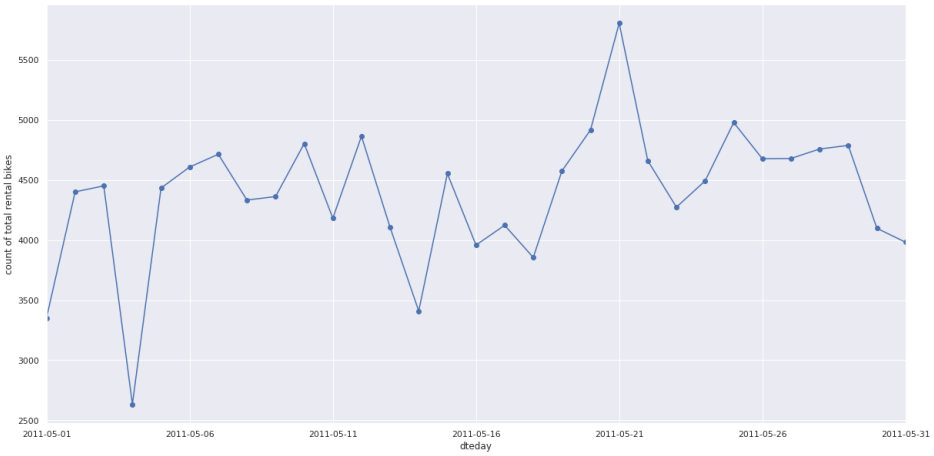

Figure 13: The count number per day in bike dataset for May.

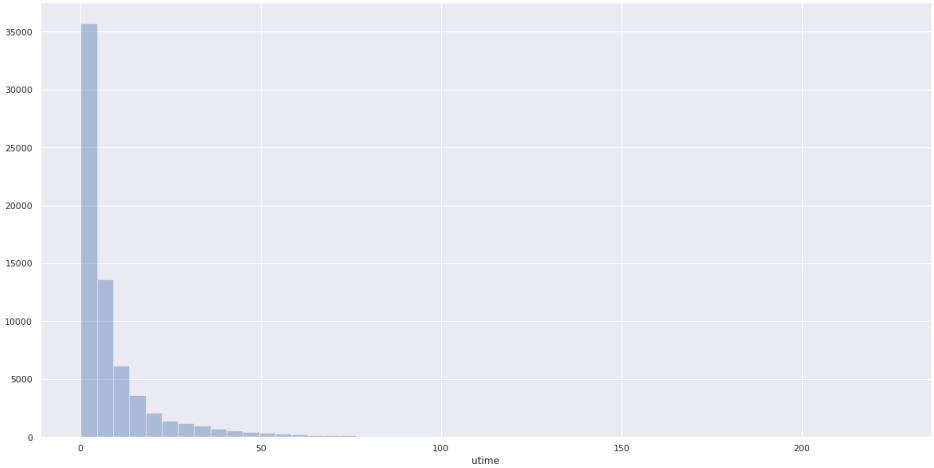

Figure 14: The distribution of video transcoding times.

mentioned in the project handout, in some cases, for example when encoding time encoding time stamps such as {Mon, ..., Sun} or {Jan, ..., Dec} it might make sense to perform either one. However, under different assumptions, the implications of one-hot encoding and scalar encoding are different. Let's say we are performing linear regression, so the assumption of one-hot encoding in this case is that there is not order or different significance between the items in that particular categorical feature, i.e. everyone is of equal significance, on the other hand the assumption of scalar encoding is that there is an order or different significance between the items in that particular categorical feature, i.e. some items have more significance than others. Thus, given this information, for the bike dataset, we decided to perform one-hot encoding to convert 'season','mnth',and 'weekday' into numbers, because there are all of equal significance. The same as in the video dataset, we decided to perform one-hot encoding to convert 'codec',and 'o_codec' into numerical vectors.

Question 7

For a data set without standardization, some features that extremely large will control the regression functions. In this part, we use *StandardScaler* function from *sklearn.preprocessing* to standardize the dada. After standardization, each columns in the data set has zero mean and unit variance, looking like a standard normally distribution.

Question 8

Mutual information (MI) of two random variables is a measure of the interdependence between the two variables. Defined as:

$$I(X;Y) = D_{KL}(\mathbb{P}_{(X,Y)}|\mathcal{P}_X \otimes \mathcal{P}_Y) \tag{8.1}$$

And $F - score$ is a measure of a test's accuracy. The F score is defined as the weighted harmonic mean of the test?s precision and recall. This score is calculated according to

$$F = \left(\frac{recall^{-1} + precision^{-1}}{2}\right)^{-1} = 2 \cdot \frac{recall \cdot precision}{recall + precision} \tag{8.2}$$

In this part, we use *mutual info* and f functions in the $sklearn.feature\_selection$ package to achieve this goal. Both of them give us the relationship between the feature and the target and help us to find the best k features. But the exact number of features (value k) is still unclear. We use $cross\_validate$ as evaluation to search the best k value in the linear regression model of MI and $F - score$ by RMSE. The impact of feature selection step on the test RMSE is shown as Figure 15 and Figure 16. One thing that needs to notice is that, for MI, the video data set is too large to calculate. Even 24 GB RAM will be overflowed. So, we only calculate the RMSE of the F score.

It shows that the RMSE will increase as k increases at first, but then it will hold stable or even drop as k continues increases. The reason is there will be overfitting will k is extremely high. Thus we can get the best k for linear regression as Table 1.

DataSet	MI	F-score	Use
Bike	19	22	20
Video	/	10	10

Table 1: Best K in linear regression without regularization

Question 8 continued on next page...

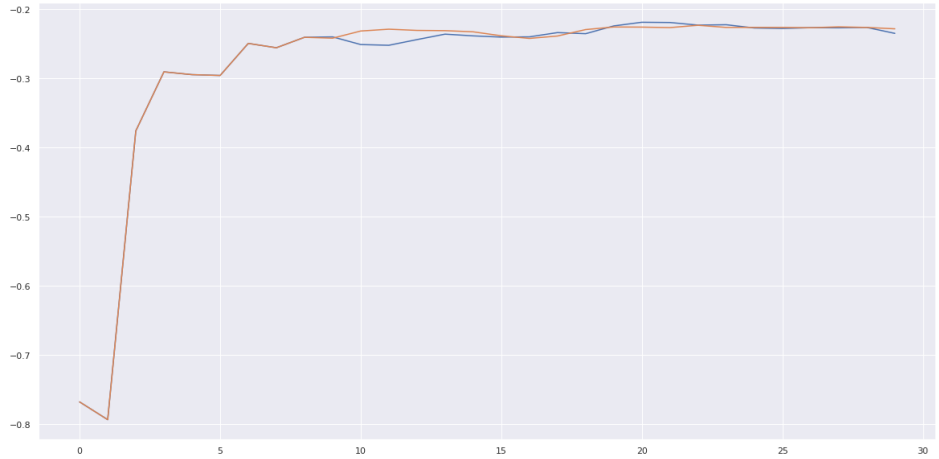

Figure 15: For MI and F score, the average RMSE against k in the bike data set.

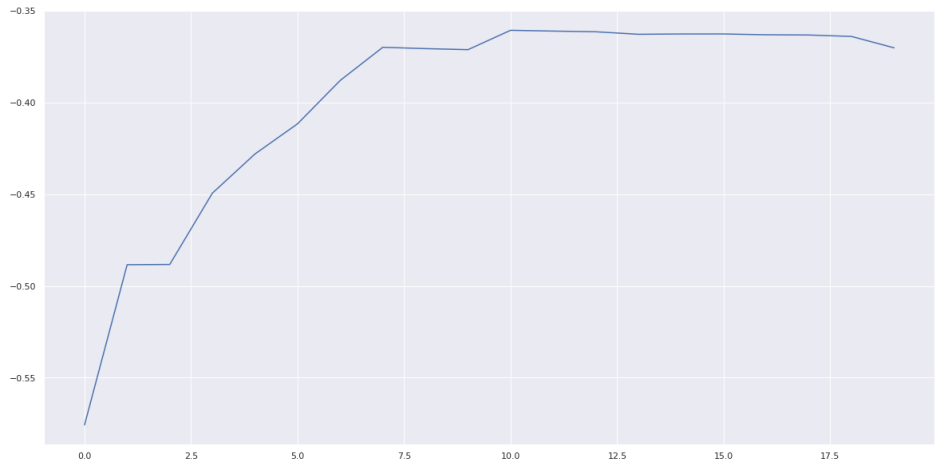

Figure 16: For F score, the average RMSE against k in the video data set.

Considering both MI and F Score, we use k=20 in the bike data set while choose k as 10 in video data set.

Question 9

First, we will demonstrate the definition of L_1 and L_2 regularization. For L_1 regularization, we have:

$$L = E_{in} + \alpha \sum_j |\omega_j|, \ \sum_j |\omega_j| \leq C \tag{9.1}$$

while L_2 regularization is defined as

$$L = E_{in} + \alpha \sum_j |\omega_j|^2, \ \sum_j |\omega_j|^2 \leq C \tag{9.2}$$

First, we will demonstrate impact of regularization strength. In the fitting process, if the parameter α is small, ω can be brought closer to the position of the optimal solution. If α is approximately 0, it is equivalent to the circular region covering the optimal solution position. At this time, regularization fails and it is easy to cause overfitting. Conversely, if α is large, the value corresponding to the above is small. At this time, the circular area is small, and ω is far away from the position of Ein's optimal solution. w is limited to change in a small area, ω is generally small and close to 0, which has the effect of regularization. However, too large α can easily cause under-fitting. Underfitting and overfitting are two opposing states.

Moreover, the impacts of different regularization on the learned model have been articulated in Project 1.

L1 regularization manage to sparsify the coefficients we are going to learn. After applying this regularization, most of the coefficient will be restricted to zero. This regularization is always applied to huge number of features training data so that we can use this regularization to remove some feature altogether, namely this works well for feature selection.This is why the performance of the model will deteriorate a little bit.

L2 regularization manage to avoid overfitting on the training set and will prefer learned weight with small norm. If regularization strength selected properly, the generalization ability of the model can be improved with the introduction of regularization as we can see through the comparison between vanilla logistic regression without regularization and model with L2 regularization. We always introduce strong L2 regularization to the complex model to avoid the model just repeat the pattern shown in training set even though such pattern may be contaminated.

Question 10

Rather than directly using $RMSE$ to compare the model performance, here we will use the score of model, which is more straightforward. We will first find the best penalty parameter for L_1 regularization and L_2 regularization using $LassoCV$ and $RidgeCV$ respectively. Then, we will compare the performance of best models derived from these two regularization and finally determine the optimal regularization scheme. The result for bike data is shown as Table 2 while the result for video data is shown as Table 3. Then the best penalty scheme for both data sets is L_2 regularization while the best penalty strength is 1.

Question 11

Data standardization is to scale the data proportionally and force them to fall into a small interval. The normalized data can be positive or negative, but the absolute value is generally not too large. The purpose of the scaling transformation for different feature sizes is to change the features of different scales to

Question 11 continued on next page...

Model	best α	score
Linear Regression (w/o regularization)	/	0.8361
Lasso Regression (L_1 regularization)	0.01	0.8328
Ridge Regression (L_2 regularization)	1.0	0.8361

Table 2: The performance comparison between different linear models for bike data.

Model	best α	score
Linear Regression (w/o regularization)	/	0.6407
Lasso Regression (L_1 regularization)	0.005	0.6403
Ridge Regression (L_2 regularization)	1.0	0.6408

Table 3: The performance comparison between different linear models for video data.

be comparable without changing the distribution of the original data. The feature scaling will definitely influence the results of the linear regression. As we have been talked in Question 8, we choose different k values, which means k most important feature as the training set X of the linear regression. In the same way, we calculate the RMSE, but this time, the data is without standardization. The result is shown as 17 and 18.

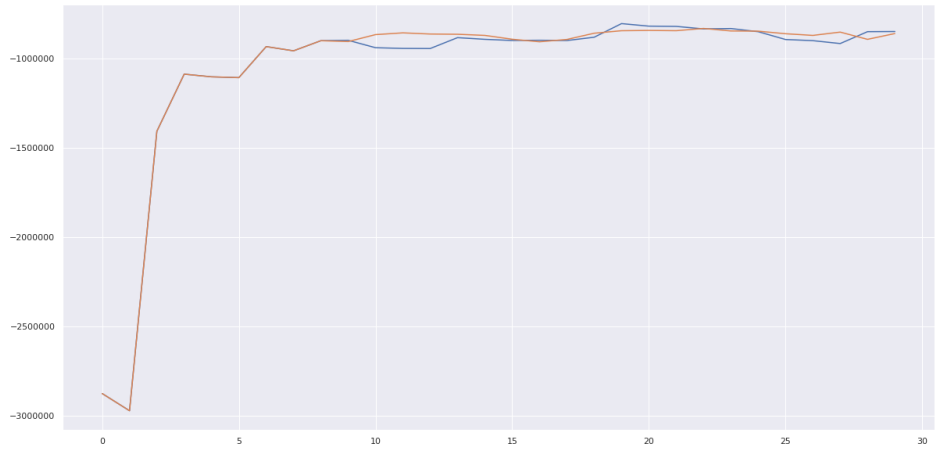

Figure 17: The trend of RMSE aganist k features selected according to F-score (with and without regularization) based on raw bike data.

Compare the result to Question 8, we find that the tendency of RMSE is the same, but the values have been extended greatly (about). As for the regularization, it does increase the performance, but not as much as standardization does.

The benefits of standardization are threefold. First, the distribution of the original data is not changed, and the influence weight of each feature dimension on the objective function is maintained. Secondly, the influence on the objective function is reflected in the geometric distribution. Finally, it is stable with enough existing samples, which is suitable for modern noisy big data scenarios.

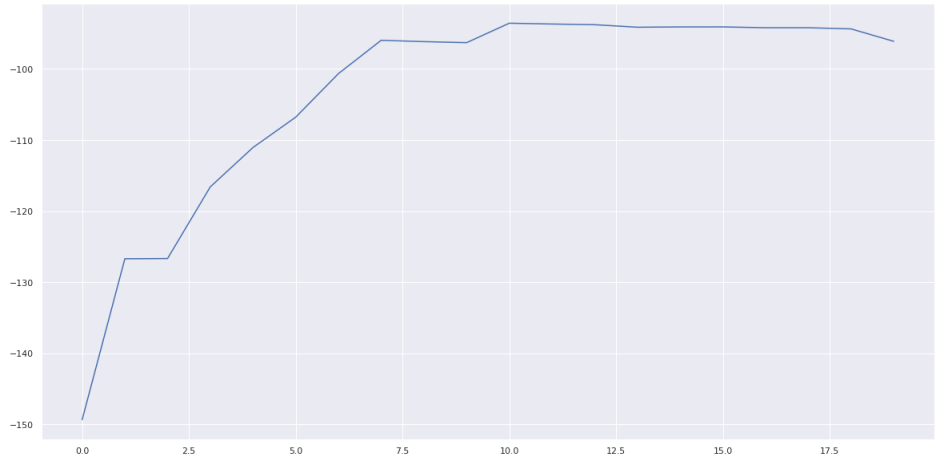

Figure 18: The trend of RMSE aganist k features selected according to F-score (with and without regularization) based on raw video data.

Question 12

According to the document, in the package *statsmodels.regression.linearmodel.OLS* , the p-value is for the t-stats of the params. And in *scipy.stats.linregress*, it is for a hypothesis test whose null hypothesis is that the slope is zero, using the Wald Test with t-distribution of the test statistic.

So in summary, we can know that the meaning of P-value in regression analysis is as follows. The p-value for each term tests the null hypothesis that the coefficient is equal to zero (no effect). A low p-value indicates that you can reject the null hypothesis. In other words, a predictor that has a low p-value is likely to be a meaningful addition to your model because changes in the predictor's value are related to changes in the response variable. Conversely, a larger (insignificant) p-value suggests that changes in the predictor are not associated with changes in the response.

The *SelectFpr* method is based on the FPR test, False Positive Rate, which means a false positive rate, which refers to the proportion of samples that we predict to be positive but actually negative, that is, the proportion of errors in a type of hypothesis test.

Here, the features are filtered according to the p-value. The smaller the p-value, the better. All features with p-values lower than the threshold alpha? we set will be selected.

However, since we used the *SelectKBest* function for feature extraction in this project, it does not involve the problem of p-values.

Question 13

We will assume the most salient features are the ones whose corresponding coefficients have high absolute values in the best regression model we find. Concrete implementation can refer to our Code about Question13 $\sim$ 14.

Based on the implementation mentioned in Question 14, for the bike data set, we find the model that performs best for validation set is the polynomial regression model with degree 2 and using L_1 regularization

with strength 0.01 with the top 13 features selected by $F-score$. The feature having largest absolute value coefficient is shown in Table 4. This result is consistent with our prior knowledge about the possible factors influencing the account of renting bikes. If the temperature is high, then less people will choose bicycle as their commutation tools. The weathersit and season both have direct association with temperature. Hence, it is alright for them to present as the most salient features. Moreover, the bike renting transaction may be more popular and better known than first year, leading to the high influence of the year on account.

For video data set, the result is similar: The polynomial regression model with degree 3 performs best after selecting the top 13 feature with highest $F-score$ accompanied with L_1 regularization of strength 0.01. The most salient features for video data set is shown at Table 4 as well. According to the table, there are three features that count. The O_{width} has direct relationship with the size of output video while the $O_{bitrate}$ has relationship with the lasting time of the output video. Meanwhile, the coding standard type also influence the size of output. Hence, these three factors are reasonable to be the most salient features.

DataSet	1_{st}	2_{nd}	3_{rd}	4_{th}	5_{th}
Bike	yr	$temp$	$temp^2$	$season_1^2$	$weathersit$
Video	$O_{width} \cdot O^2_{codec_{h264}}$	O_{width}	$O_{bitrate} \cdot O^2_{codec_{h264}}$	$O^3_{codec_{h264}}$	$O_{width}O_{bitrate}O_{codec_{h264}}$

Table 4: The top salient feature for bike and video data set based on the absolute value of their coefficients.

Question 14

In this answer, we will demonstrate the hyper-parameters that may affect the performance of our models, the practical restriction that hinder us increasing the polynomial degree and how these factors affect our strategy to find the best parameters setting.

There are three parts of our pipeline. The first part is the feature selection model which picks up the top k features according to $F-score$ or MI. The second part is the polynomial transformer, which construct the polynomial features with wanted degree. The last part is the linear regression model with different regularization, namely the Lasso Regression model and Ridge Regression model. According to construction law of polynomial features, if we have n original features and will apply d degree of polynomial transformation, we will finally have

$$\frac{(n+d)!}{n!d!} = \binom{n+d}{d} \tag{14.1}$$

features. In order to avoid the explosion of feature dimension, we will first find the optimal number of features selected and corresponding feature selection metric based on the result of polynomial regression of degree 2 with fixed regularization. Then we will determine the best polynomial degree, finally the best regularization form and corresponding strength.

The impact of feature selection can be seen through our first step of greedy search as well. Taking the bike data set as an example, the impact of feature selection on the $RMSE$ is shown as Table 5, which is consistent with the trend we guess in Question 8.

Degree higher than 5 is not considered here because 5 is sufficiently high for us to predict the trend of performance with increasing degree based on the existing grid searching result. Moreover, because of the large size of video data set, we restrict the maximum degree of polynomial regression as 3 to avoid unacceptable computation resource demand. According to a series of greedy decisions, we find the polynomial regression

Question 14 continued on next page...

Top features	F-score	Mutual Information
	RMSE	RMSE
1	1.0416	0.8817
4	0.3148	0.3148
7	0.3142	0.2811
10	0.3227	0.3517
13	0.2656	0.3685
16	0.2711	0.3894
19	0.3678	0.3789
22	0.3091	0.3902
25	0.3266	0.3469
28	0.3826	0.3070
31	0.3331	0.3331

Table 5: The average RMSE for different strength of feature selection and metrics based on bike data.

model with degree 2 performs best among 4 choice of degrees on bike data set. The optimal degree of polynomial regression for video data set, however is 3.

In all, there are two main reasons that prevent us from increasing the polynomial degree infinitely. First reason is from the practical consideration about computation resources as mentioned. Increasing polynomial degree will result in the exponential increment of computation for the optimization. Secondly, the high degree of polynomial regression will cause the increasing model capacity, which will induce large generalization error even with small training error achieved.

Question 15

The reason that we need to craft such features is that the polynomial regression can never fit the inverse relationship between target and features. Based on our intuition about the trans-coding, there are several factors impacting the speed of it such as the lasting time of the whole video, size of video, bit rate of video. A reasonable crafting of new feature should be $\frac{1}{O_{bitrate}}$ since the higher the bit-rate is, the less lasting time the whole video will have. The $F-score$ of inverse bit-rate, which are much higher than the one of bit-rate, also evidence our intuitions. We are going to concatenate this feature with the original input as the input of our optimal model obtained in Question 14. The comparison between the original result and the RMSE of this method is shown as Table 6. According to the result, the crafted feature does remain after feature selection and helps the model predict the entire transcoding time precisely.

Metric	Crafted Feature	Original
RMSE	0.1031	0.1057

Table 6: The performance comparison between original data and crafted data.

Question 16

From the view of model complexity, there are at least two factors that may cause the better performance of neural networks. First of all, the number of parameters of neural networks is always much larger than the

Question 16 continued on next page...

one for linear regression, leading to better ability to fit the data with small generalization error if adopted proper implicit regularization. Secondly, the activation for the hidden units are non-linear functions, which has more flexibility even though the number of parameters is same as the linear model.

Question 17

We will construct a pipeline before going to find the best combination of hyper-parameters. The pipeline just includes the MLP model. We will provide 4 possible choices of the network depth which are $1, 2, 3, 5$ respectively, 3 alternatives for the number of hidden neurons at each layer, which are $30, 50, 60$ respectively, and all the classical activation functions. No feature selection is adopted here because the number of input features affects the model complexity and the computation resources requisite subtly. Moreover, we will keep the number of neurons of each layer is the same for the whole neural network. Details can be seen from our code for Question 17.

In all, for these two data set, the best hyper-parameter combination is shown as Table 7 while the corresponding average test R^2 score is shown as well. Concrete performance of each combination of hyper-parameters can refer to the two excel files, $bike\_MLP\_result.xls$ and $video\_MLP\_result.xls$.

Dataset	Layers	Hidden Neurons/Layer	Activiation	Mean R^2 score
Bike	3	30	Logistic	0.3670
Video	5	50	ReLu	0.8680

Table 7: Best hyper-parameter settings and corresponding R^2 score for different data set.

Question 18

The best activation function for regression should be identity. First of all, most of activation functions have limited range which restricts their application to practice. For example, the output of $tanh$ is limited to interval $(0, 1)$ while the output of $ReLu$ is limited to $(0, \infty)$. The identity projection, however covers the whole real domain. Moreover, we should notice that the neural network itself can nearly approach any possible function. Even though our target does fall into the range of one specific activation function, it is still possible for neural networks to fit these data without any output activation having sufficient neurons. In all, for regression, identity projection should always be adopted.

Question 19

Notice that for neural networks, both sizes of bike data and video data are much less comparing to the data will be processed for practical application, such as ImageNets. Hence, there is no need for the consideration about the computation resources. The most important factor that preserves us from increasing the depth of the network is the trade-off between model complexity and generalization ability. Because the implicit regularization like weight decay cannot fully restrict the model capacity, using models with increasing number of parameters will result in over-fitting to the training data. Another possible reason for limited network depth can be seen in the paper of ResNet, which shows that the non-linear layer of neural networks cannot perform identical projection properly, leading to worse performance of model with more layers introduced.

Question 20

The maximum number of features can influence the generalization error in two ways. If we have many features, it will increase the strength of each individual tree. If we have less features, it will lead to a lower correlation among the trees, but increase the strength of the entire forest.

For the number of trees, the more we have, the better result we should get. However, the improvement gets smaller and it could eventually overfit for some dataset. There is a suggestion for number of trees to be 64 ? 128. It will reach a balance between error rate and processing time.

For the depth of the tree, the deeper it is, the more splits it has and it is supposed to capture more information about the data. However, once it reaches a certain depth, it will start overfitting and could not generalize for test datasets.

We found the best value by trying within a range. The result we found was with a max depth of 14, max features of 9, n_estimators of 50.

Question 21

Random forest consists of a large number of individual decision trees. It works as an ensemble that counts the vote of each decision tree. The class with the most votes is the model?s prediction value. The uncorrelated individual trees operate together to cover each other?s errors, thus it works better than any individual models. It is basically the power of teamwork.

Question 22

One of the tree in our random forest model is shown as Figure 19. The feature used for branching at the root node is the most important feature. This way, the most effective feature is used first to eliminate the bad options. Then, feature is used in the order of their importance. The order of importance is very similar to the order we inspected during linear regression. But because they are evaluated using different methods, it is slightly different.

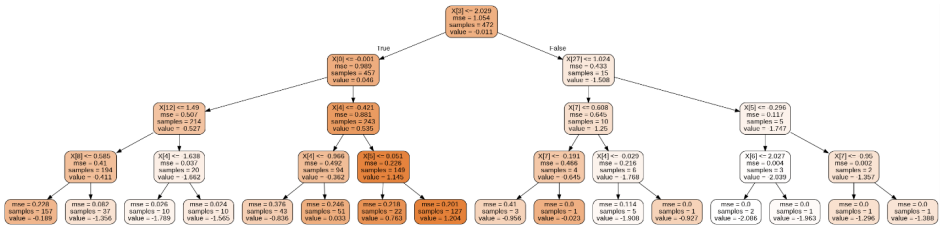

Figure 19: One of the tree in derived random forest model.

Question 23

We have performed 10-fold cross-validation and measured average RMSE errors for training and validation sets for each model. For the specific RMSE value of each model, please refer to the relative question, we here just exhibit the RMSE of random forest model as Table 8.The training RMSE error is usually lower than that of the validation sets, simply because the model is trained on these data and learned these data?s

Question 23 continued on next page...

specific characteristics. It sometimes may overfit and result in much higher error rate in the validation data sets, because the model learned too specific and not general enough.

DataSet	*Average Training RMSE*	*Average Validation RMSE*
Bike	0.1380	0.4695
Video	0.5128	1.8944

Table 8: The performance of Random Forest Model on training set and validation set.

Question 24

The OOB score for bike data set is 0.8902 while the OOB score for video data set is 0.5349.

R^2 score is used for any regression models to evaluate its goodness of fit. However, having a high R^2 in the training dataset doesn?t necessarily mean anything. On the contrary, out of bag error is used specifically for random forest because of its bagging behavior. It is the best way of validating the random forest model.

Each of the out-of-bag sample rows is put through the decision trees that do not contain the out-of-bag sample row in its bootstrap training data and a major prediction is used. So, the out-of-bag error is calculating the correctly predicted rows from the out of bag samples. Compared to the R^2 score, OOB is calculated based on data that was not part of the analysis of the model. To calculate a validation score, we have to set aside part of the data before training the model. Also, OOB score is calculated based on only a subset of decision trees that do not contain the OOB sample in their bootstrap. But the R^2 is calculated using all the decision trees. They are two completed different approaches towards evaluation of the model.