Question 1:

Our approach to the **validation** method was to vary the ratio (or split) of the **train subset** to the **test subset** (of the training data), in a sense causing the ratio to become a hyperparameter of the model, while fixing the ratio of the **test** subset to the **validation** subset to exactly 50% (0.5). For each train subset–to–test subset ratio [**0.2, 0.3, 0.5**], the (**simple**) **Ordinary Least Squares** model was trained on the “**train + validation**” subsets (**model selection**). The model that yielded the lowest RSE was then used on the **test** subset (**model assessment**).

Displayed below and tabularized are sample scores from an instance of running validation on the training dataset.

```
Simple Linear Regression
x_train_sub: 160, y_train_sub: 160
x_val_sub: 320, y_val_sub: 320
Validation:
RSE_val_sub: 10.437398373022999
RSquare_val_sub: 0.6173016047616007
----------------------------------------
x_train_sub: 240, y_train_sub: 240
x_val_sub: 280, y_val_sub: 280
Validation:
RSE_val_sub: 10.555812882473376
RSquare_val_sub: 0.5890871793136658
----------------------------------------
x_train_sub: 400, y_train_sub: 400
x_val_sub: 200, y_val_sub: 200
Validation:
RSE_val_sub: 10.647141688182703
RSquare_val_sub: 0.5925505876774707
----------------------------------------
x_train_sub: 480, y_train_sub: 480
x_test_sub: 320, y_test_sub: 320

Test:
RSE_test_sub: 10.399987291453957
RSquare_test_sub: 0.6406625212834421
```

train : validation : test	**RSE**	**R^2**
0.2 : 0.4 : 0.4	10.4374	0.6173
0.3 : 0.35 : 0.35	10.5558	0.5891
0.5 : 0.25 : 0.25	10.6471	0.5926

Figure 1 – Sample results of validation

For **cross-validation**, **model selection** was carried out by varying the hyper-parameter K, the number of folds created from the training dataset, and consequently the number of samples held out from training the model at a given moment. The average RSE and R^2 values of the model's performance for each K were calculated. Subsequently, the model with the lowest RSE (optimal hyper-parameters) was fit on the entire training dataset and run on the test set to evaluate its performance (**model assessment**).

```
Simple Linear Regression : [K=2]
Cross Validation(n_splits=2):
RSE_cv_fold: 10.591530975350555
RSquare_cv_fold: 0.607473200454563
========================================
Simple Linear Regression : [K=3]
Cross Validation(n_splits=3):
RSE_cv_fold: 10.432356065853336
RSquare_cv_fold: 0.6205865700222323
========================================
Simple Linear Regression : [K=5]
Cross Validation(n_splits=5):
RSE_cv_fold: 10.382679026538245
RSquare_cv_fold: 0.6268905748544279
========================================
Simple Linear Regression : [K=10]
Cross Validation(n_splits=10):
RSE_cv_fold: 10.4683512131033
RSquare_cv_fold: 0.615260325225556
========================================
Simple Linear Regression : [K=15]
Cross Validation(n_splits=15):
RSE_cv_fold: 10.514479535044872
RSquare_cv_fold: 0.6094834518910771
========================================
Simple Linear Regression : [K=20]
Cross Validation(n_splits=20):
RSE_cv_fold: 10.605818795201769
RSquare_cv_fold: 0.6014191871018241
========================================
Simple Linear Regression : [K=30]
Cross Validation(n_splits=30):
RSE_cv_fold: 10.666731981667208
RSquare_cv_fold: 0.5936544396949841
========================================
Simple Linear Regression : [K=50]
Cross Validation(n_splits=50):
RSE_cv_fold: 10.836096600621138
RSquare_cv_fold: 0.5574515576671456
========================================
Simple Linear Regression : [K=100]
Cross Validation(n_splits=100):
RSE_cv_fold: 11.519408984614506
RSquare_cv_fold: 0.38673367020132643
========================================
RSE_cv_test: 9.638450941285175
RSquare_cv_test: 0.5953110867026103
```

K	RSE	R^2
2	10.5915	0.6075
3	10.4324	0.6206
5	10.3827	0.6269
10	10.4684	0.6153
15	10.5145	0.6095
20	10.6058	0.6014
30	10.6667	0.5937
50	10.8361	0.5575
100	11.5194	0.3867

Figure 2 – Sample results of cross-validation

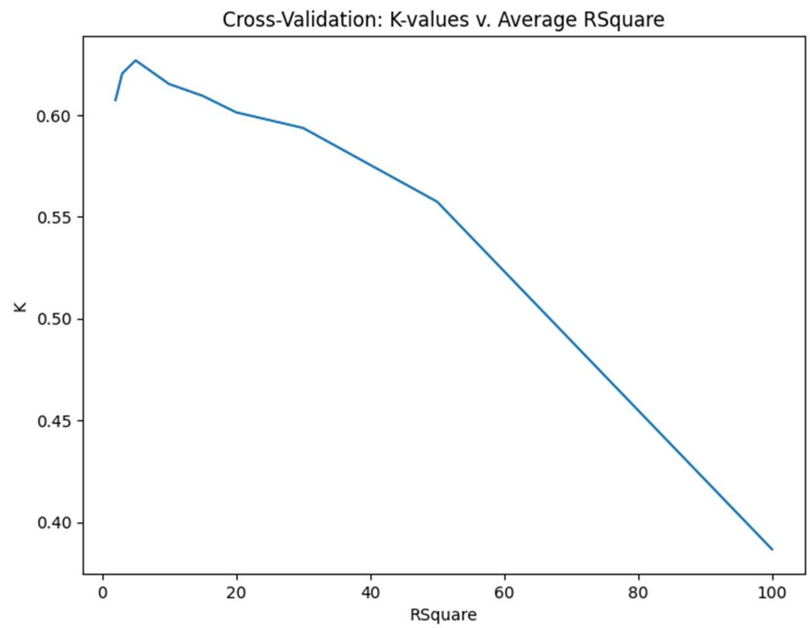


Figure 3 – K-values v. Average R^2 (cross-validation)

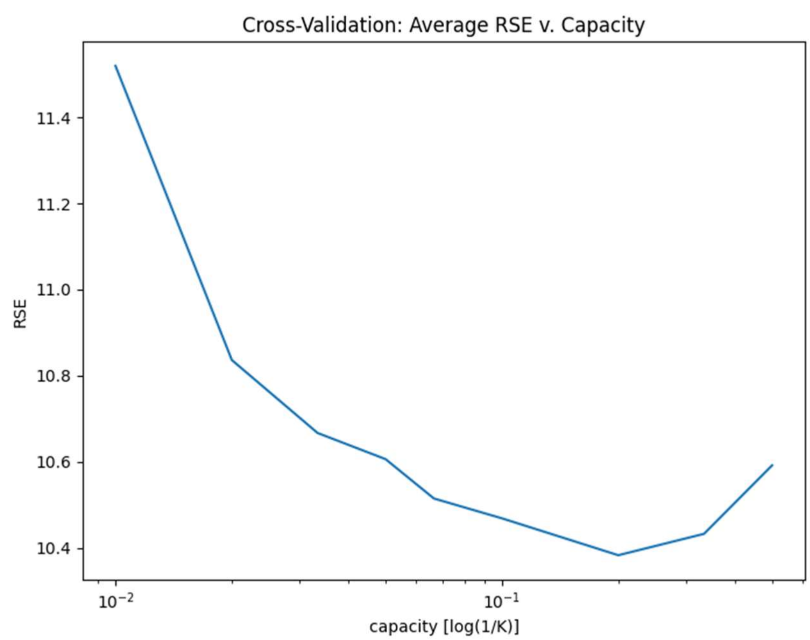


Figure 4 – Average RSE v. Capacity (cross-validation)

In Figure 3, as expected, the R^2 peaks around K-values of 3, 5, 10 and worsens as K increases; knowing that **smaller** values of K are preferred for cross-validation when the size of the dataset, N, is **large(r).** Conversely, in Figure 4, the RSE decreases as the capacity increases.

From the same run of validation, and cross-validation, below are the results of the model assessment; for validation, running the optimal model obtained from training on "**train + validation**" subsets on the **test** subset (of the training dataset), and for cross-validation, running the optimal model obtained, from the various K-values, on the actual **test** dataset.

Method	**RSE**	**R^2**
Validation	10.4000	0.6406
Cross-Validation	9.6385	0.5953

Figure 5 – Comparison of assessment of Validation and Cross-Validation methods

As observed above, the performance of the optimal model using the validation method on the "**test**" subset of the training data surpasses the performance of the optimal model obtained using K-fold cross-validation on the actual (blind) test dataset.

Question 2:

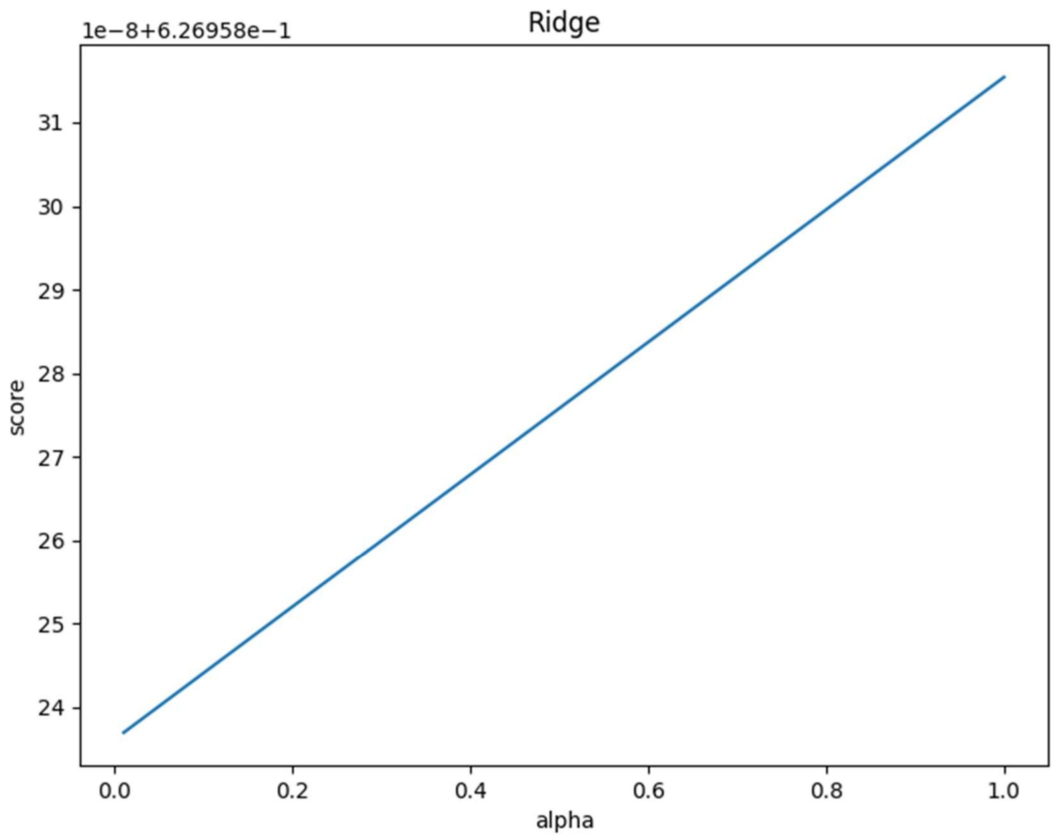
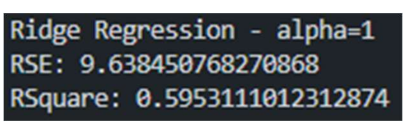


Figure 6 – alpha values v. scores of Ridge Regression Model



Method	RSE	R^2
Simple OLE Validation	10.4000	0.6406
Simple OLE Cross-Validation	9.6385	0.5953
Ridge Regression	9.6385	0.5953

Figure 7 – Comparison of assessment of Simple OLE and Ridge Regression

As observed in Figure 6, the model performance of the Ridge Regressor is directly proportional to the alpha hyper-parameters, with the performance peaking at the highest alpha value during cross-validation.

Comparing the RSE and R^2 values of the Ridge Regressor using the optimal alpha value of 1, we observed approximately (to 4 d.p.) the same performance as the Simple OLE (cross-validation), but slightly worse than the Simple OLE validation (**NOTE:** Simple OLE validation was not on the actual test dataset, but on test subset of training dataset).

Question 3:

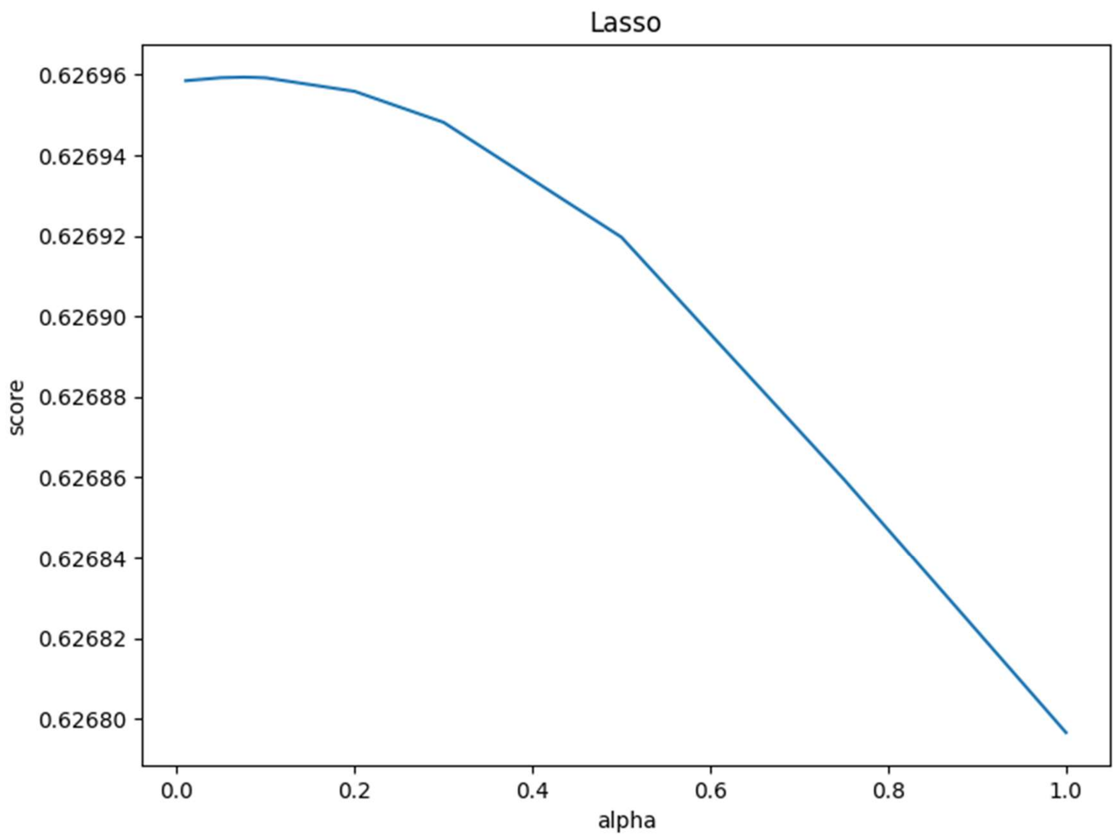


Figure 8 – alpha values v. scores of Lasso Regression Model

```
Lasso Regression - alpha=0.075
RSE: 9.639263910631346
RSquare: 0.5952428156581796
```

Method	**RSE**	**R^2**
Simple OLE Validation	10.4000	0.6406
Simple OLE Cross-Validation	9.6385	0.5953
Ridge Regression	9.6385	0.5953
Lasso Regression	9.6393	0.5952

Figure 9 – Comparison of assessment of Simple OLE and Ridge Regression

As observed in Figure 8, in contrast to the Ridge Regressor, the model performance of the Lasso Regressor is inversely proportional to the alpha hyper-parameters, and actually peaks somewhere around the lowest alpha value (expected based on how it optimizes the objective function).

Comparing the RSE and R^2 values of the Lasso Regressor using the optimal alpha value of 0.075, we observed an extremely miniscule decrement in model performance to the Simple OLE and the Ridge Regressor.