5 Model Selection

Exercise 5.1

Consider the following snippet of code taking as input an `N` $\times$ `M` matrix `X` of data points and an `N`-dimensional vector `y` of binary targets, where `N` is the number samples, `M` is the number of features.

```
1  import numpy as np
2  classifier = LogisiticRegression()
3  classifier.fit(X, y)
4  yhat = classifier.predict(X)
5  accuracies = [sum(yhat = = y) / N]
6  for i in range(M):
7     Xi = np.delete(X, i, axis=1)
8     classifier.fit(Xi, y)
9     yhat = classifier.predict(Xi)
10    accuracies.append(sum(yhat = = y) / N)
```

Are the subsequent statements true or false? Provide motivations for your answers.

1. This snippet of code implements a portion of a well-known model selection procedure.

2. After having run the reported snippet, one should keep the classifier that led to the maximum value in the list of accuracies.

3. The classifier trained at line 3 is likely to suffer a larger bias than the classifiers trained at line 8.

4. The time of computation we need to run the snippet of code scales linearly with `M`.

Exercise 5.2

Answer to the following questions regarding feature selection. Provide motivation for your answers.

1. You have been asked to implement a feature selection process on a system with very limited computational resources. Would you opt for a filter approach or for a wrapper approach?

2. You have been asked to implement a feature selection process to improve as much as possible the performance of your model. Would you opt for a filter approach or for a wrapper approach?

3. If you want to rank (in order of importance) the features of a classification problem, which kind of feature selection process would you use among the ones presented in the course?

4. You trained two models on a problem with 5 features: Model A using all the 5 features and Model B using only 3 features. Assuming they have similar performances on the training set, do you expect Model A to perform better on the test set?

5. You trained two models on a problem with 9 features: Model A using only 5 features and Model B using all the 9 features. Do you expect Model A to have a smaller training error than Model B?

Exercise 5.3

We selected the features of a linear regression model by running a forward feature selection procedure which minimizes the validation error. For each of the following, tell if the elements corresponding to the learned model increase or decreases as the mode is increasing the number of feature (given that we trained on the same data):

1. The sum of the squared residuals on the same training set;

2. The variance;

3. The squared bias;

4. The sum of the squared residuals on a test set.

Exercise 5.4

Assume you are solving a regression problem with a linear regression model and you are considering to switch either to LASSO or Ridge regression. Tell which one you would choose basing on the following requirements (one at a time):

1. Too few data w.r.t. the number of parameters we are using for regression;

2. The final model is hard to interpret due to the large number of input features that have been used for the regression;

3. Bad conditioning of the design matrix $\Phi(x)^\top \Phi(x)$;

4. Problem with model bias of the current model.

Motivate your choices.

Exercise 5.5

Consider the following snippet of code taking as input an `N` × `M` matrix `X` of data points and an `N`-dimensional vector `y` of binary targets, where `N` is the number samples, `M` is the number of features. Are the subsequent statements true or false? Provide motivations for your answers.

```
1  import numpy as np
2  classifier = LogisiticRegression()
3  classifier.fit(X, y)
4  yhat = classifier.predict(X)
5  accuracies = [sum(yhat = = y) / N]
6  for i in range(M):
7    Xi = np.delete(X, i, axis=1)
8    classifier.fit(Xi, y)
9    yhat = classifier.predict(Xi)
10   accuracies.append(sum(yhat = = y) / N)
```

1. This snippet of code implements a portion of a well-known model selection procedure. (*Note*: in any case, specify the procedure and briefly describe how to complement the code).

2. After having run the reported snippet, one should keep the classifier that led to the maximum value in the list of accuracies.

3. The classifier trained at line 3 is likely to suffer a larger bias than the classifiers trained at line 8.

4. The time of computation we need to run the snippet of code scales linearly with `M`.

Exercise 5.6

Consider the following snippet of code:

```
1  coef = []
2  mse = []
3  for alpha in [0.001, 0.01, 0.1, 0.2, 0.5]:
4    model = linear_model.Lasso(alpha=alpha)
5    model.fit(x, y)
6    mse.append(mean_squared_error(y, model.predict(x)))
7    coef.append(model.coef_)
8  i = np.argmin(mse)
9  print(i, mse[i], coef[i])
```

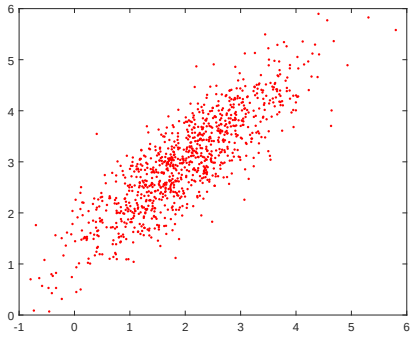
1. Describe the procedure provided by the snippet above. Is it correct? If not, propose a correction.

2. What is the purpose of the procedure above? Can you tell which property have the coefficients `coef` computed in the snippet above?

3. Can you list at least two other approaches that can be used for purposes similar to the one provided by the procedure above? Motivate your answers.

Exercise 5.7

Consider the following dataset:



Draw the direction of the principal components and provide an approximate and consistent guess of the values of the loadings. Are they unique?

Exercise 5.8

Consider the following statement regarding PCA and tell if they are true or false. Provide motivation for your answers.

1. Even if all the input features are on very similar scales, we should still perform mean normalization (so that each feature has zero mean) before running PCA.

2. Given only scores t_i and the loadings W, there is no way to reconstruct any reasonable approximation to x_i.

3. Given input data $x_i \in \mathbb{R}^d$, it makes sense to run PCA only with values of k that satisfy $k \leq d$.

4. PCA is susceptible to local optima, thus trying multiple random initializations may help.

Exercise 5.9

Which of the following is a reasonable way to select the number of principal components k in a datset with N samples?

1. Choose k to be 99% of N, i.e., $k = \lceil 0.99N \rceil$;
2. Choose the value of k that minimizes the approximation error $\sum_{i=1}^{N} ||x_i - \hat{x}_i||_2^2$;
3. Choose k to be the smallest value so that at least 99% of the variance is retained;
4. Choose k to be the smallest value so that at least 1% of the variance is retained;
5. Identify the elbow of the cumulated variance function.

What changes if the purpose of PCA is visualization?

Exercise 5.10

Consider the following snippet of code taking as input a dataset `X` having 100 samples with 5 features. Are the subsequent statements true or false? Motivate your answers.

```
1 import numpy as np
2 X_tilde = X - np.mean(X, axis=0)
3 C = np.dot(X_tilde.T, X_tilde)
4 eigenvalues, W = np.linalg.eig(C)
5 T = np.dot(X_tilde, W[:,:2])
```

1. The code snippet above is implementing a feature selection technique.
2. Line 2 is unnecessary if the data in `X` are scaled.
3. A model trained with the inputs `T` is likely to display a lower bias than a model trained with inputs `X`.
4. It is possible to recover, by computing $X = W[:, :2]^\top \cdot T$, the original dataset `X` from `T`.

Exercise 5.11

Consider the following statement regarding **Principal Component Analysis** (PCA) and tell if they are true or false. Provide motivation for your answers.

1. PCA might get stuck into local optima, thus trying multiple random initializations might help;
2. Even if all the input features have similar scales, we should still perform mean normalization (so that each feature has zero mean) before running PCA;

3. Given the scores $\tilde{x}_i, \forall i$ and the loadings matrix W, there is no way to reconstruct the original samples $x_i, \forall i$;

4. PCA can used either for data compression, data visualization or feature extraction.

Exercise 5.12

Tell if the following statement about the Principal Component Analysis (PCA) procedure are true or false. Motivate your answers.

1. The set of the Principal Components vectors are providing an orthonormal base for the original feature space.

2. Using as features for regression/classification problems the projection of the original features into the principal components provided by the PCA reduces the phenomenon of overfitting.

3. The percentage of the variance explained by a Principal Component is inversely proportional to the value of the corresponding eigenvalue.

4. The procedure to apply PCA to a dataset is deterministic.

Exercise 5.13

Assume you have been given a dataset with input matrix X and target vector y. Consider the following snippet of code:

```
1  pca = PCA()
2  pca.fit(X, y)
3  explained = pca.explained_variance_
4  T = pca.transform(X)
5  explained_variance = np.cumsum(explained) / sum(explained)
6  T_tilde = T[:, explained_variance < 0.95]
```

1. Describe the procedure and the purpose of the above code snippet. Is it correct?

2. Line 6 consists in a selection procedure. Explain the rationale behind this operation and suggest other viable options to perform the selection procedure.

3. Do you think this code requires some preliminary operations on X and y before being executed? In the case of a positive answer, tell which ones and why should they be performed. In the case of a negative answer, motivate adequately.

Exercise 5.14

Imagine having a dataset with a small amount of data you suspect is corrupted. Moreover, you have a specific application that allows you to consider only simple models in the system's training and operational life (e.g., embedded microcontroller).

1. Propose a solution for the above-described setting.
2. Instead, assume to have a long time for training. What techniques would you consider to reduce the influence of these corrupted samples on the used model?
3. Does the prediction phase of the learner increase with the proposed technique? By how much?

Exercise 5.15

Describe the **advantages** and **drawbacks** of the following choices about model selection in Machine Learning:

1. Increase the model complexity and fix the number of samples;
2. Fix the model complexity and the number of samples, but use ensemble techniques.

Exercise 5.16

State whether the following claims about Bagging and Boosting are true or false, motivating your answers:

1. Since Boosting and Bagging are ensemble methods, they can be both parallelized.
2. Bagging should be applied with weak learners.
3. The central idea of Boosting consists in using bootstrapping.
4. It is not a good idea to use Boosting with a deep neural network as a base learner.

Exercise 5.17

Tell which technique or approach would you use for the following purposes. Motivate your choice.

1. Reduce the variance of a model;
2. Select a model without retraining it on a different set of data;
3. Reduce the bias of the model, without increasing its variance;
4. Select a model by exploiting the huge computational power available to you.

Exercise 5.18

Answer to the following questions about the bias-variance decomposition, model selection, and related topics. Motivate your answers.

1. If your linear regression model underfits the training data (i.e., the model is not complex enough to explain the data), would you apply PCA to compute a more suitable feature space for your model?

2. If solving a regression problem, the design matrix $X^\top X$, is singular, would you apply PCA to solve this issue?

3. Assuming a classifier fits very well the training data but underperforms on the validation set, would you apply Bagging or Boosting to improve it?

4. Assuming that you trained a classifier with a K-fold cross-validation and it consistently has poor performances both on training and on validation folds, would you apply Bagging or Boosting to improve it?

5. You applied ridge regression to train a linear model using a rather large regularization coefficient, would you think that bagging would improve your model?

6. You have been asked to implement a feature selection process on a system with very limited computational resources. Would you opt for a filter approach or for a wrapper approach?

7. You have been asked to implement a feature selection process to improve as much as possible the performance of your model. Would you opt for a filter approach or for a wrapper approach?

8. You need to train a linear regression model using as input the readings of several sensors. Assuming that you know that some of these sensors might be faulty (i.e., resulting in meaningless readings), which linear regression approach would you use to train your model?

9. A linear regression model, computed using ordinary least squares, has a validation error that is much larger than training error. Assuming that you do not want to change neither the input features nor the kind of model, what would you do to improve it?

10. If you have to choose among a few models knowing only the training error (assuming you cannot retrain them or evaluate them on a different dataset), what would you do?

Exercise 5.19

Answer to the following questions about the bias-variance decomposition, model selection, and related topics. Motivate your answers.

1. You trained two models on the same dataset on a problem with 5 features: Model A using all the 5 features and Model B using only 3 features. Assuming they have similar performances on the training set, do you expect Model A to perform better on the test set?

2. You trained two models on the same dataset on a problem with 9 features: Model A using only 5 features and Model B using all the 9 features. Do you expect Model A to have a smaller training error than Model B?

3. You trained a K-NN classifier and the performance on the validation set is much worse than the one on the training set. Would you increase the value of K?

4. You trained a model with ridge regression and the performance on the validation set is much worse than the one on the training set. Would you decrease the regularization coefficient?

5. You used 10-fold cross-validation to tune the hyper-parameter K of a classifier. The model trained on the third fold with $K = 3$ achieved the best performance overall. Based on this, would you set $K = 3$?

6. You used 10-fold cross-validation to select a classification model among several ones. Once you selected the model, would it be a good idea to re-train it on the whole dataset (i.e., all the 10 folds together)?

7. You trained 10 regression models applying different basis functions to the problem inputs. Assuming you do not have enough time to perform additional training, would you select the model with the lowest training error?

8. You need to assess the performance of a model on a very large dataset. You discover that training your model on the whole dataset is not very (computationally) expensive. Would you use Leave-one-out (LOO) cross-validation?