Homework 5

Yixuan Li
CSE 151A: Machine Learning and Algorithms

May 8, 2025

1 Conceptual and Mathematical Problems

1. *Regression with d + 1 data points in d dimensions.*

$$
\begin{aligned}
&wx^{(i)} + b = y^{(i)} = c_i \\
&x^{(0)} : c_0 = b \\
&x^{(1)} : w_1 + b = c_1 \Rightarrow w_1 = c_1 - c_0 \\
&x^{(2)} : w_2 + b = c_2 \to w_2 = c_2 - c_0 \\
\dots \\
&x^{(d)} : w_d + b = c_d \to w_d = c_d - c_0
\end{aligned}
$$

$$
w = \begin{bmatrix} c_1 - c_0 \\ c_2 - c_0 \\ \dots \\ c_d - c_0 \end{bmatrix}
$$

$$
b = c_0
$$

2. *Effect of regularization in regression.*

(a) When $\lambda = 0$: the problem is equal to a regular least square regression. So, since we can find w, b that perfectly fit these points, so we can fit a perfect model with training loss equals 0:

$$
L_0 = 0
$$

(b) As λ increases, $\|w_\lambda\|$ will decrease.

(c) As λ increases, L_λ will increase.

(d) As λ goes to infinity, $\|w_\lambda\|$ will approach zero, because our aim is to minimize the training loss. So, the term $w_\lambda x^{(i)}$ in the training loss will be zero, and L_λ will approach $\sum_{i=0}^{d}(c_i - \frac{\sum_{i=0}^{d} c_i}{d+1})^2$.

3.

(a) No. There is not likely to be a significant amount of inherent uncertainty, because there should always exist a certain name of an animal.

(b) Yes. There exists a significant amount of inherent uncertainty because there are a lot of uncertainties that two people will be interested in each other or not.

(c) No. There is not likely to be a significant amount of inherent uncertainty, because the speech is always certain, so the transcription of it will not vary a lot.

(d) Yes. There exists a significant amount of inherent uncertainty because there are many factors that determine whether a new song will be a big hit or not.

4.

(a) Model (w', b') makes 2 mistakes, and model (w'', b'') makes 1 mistake.

(b) The logistic loss of model (w', b'): $L(w', b') = 1.93$.

The logistic loss of model (w'', b''): $L(w'', b'') = 10.36$.

(c) In part (a), we only calculate the number of mistakes the two models made on the training set, and it seems like model (w'', b'') makes fewer mistakes than (w', b').

However, in part (b), we can tell model (w', b') is better than model (w'', b'') because the former has smaller logistic regression loss. The big loss of model (w'', b'') is mainly because the model wrongly predict the second index with over-confidence, which can generate a huge loss.

So, the discrepancy shows that logistic loss is more sensitive to the over-confidence on wrongly predicted data. Since error only cares about the prediction result, sometimes it is normal that the model with fewer logistic loss makes more errors.

5.

(a) As m grows, $f(m)$ will decrease.

(b) As m grows, we expect $e(m)$ to decrease, but it will not necessarily behave in this way, and sometimes it will temperately increase a little bit. This may be because of the slight shift of decision boundary during the learning process of the model.

2 Programming Problems

6. *Binary logistic regression.*

(a) The coefficient of the model is shown in Fig. 1:

(b) The three most influential feature that I choose is: `ca`, `thal` and `sex`. I choose these features based on the absolute value of the coefficients, and select the biggest Top 3.

(c) The test error of the model is 0.1942.

(d) The error by 5-fold cross validation on the training set is 0.1500.

Compared to the test error, the 5-fold cross validation error is slightly smaller than test error, which is normal. It shows that the model may be slightly underfitting or not scalable enough. Because the test data set is completely new to the model, and on 5-fold cross validation, the model can be access to the same parts of the training data every time.

```
Model coefficients:
age: 0.0219
sex: -1.0896
cp: 0.8885
trestbps: -0.0085
chol: -0.0009
fbs: 0.1406
restecg: 0.4980
thalach: 0.0184
exang: -0.8572
oldpeak: -0.4815
slope: 0.7509
ca: -1.3712
thal: -1.2199
```

Figure 1: Problem 6 (a): Coefficient of the model

7. *Stepwise forward selection.*

(a) The code of the stepwise forward algorithm `stepwise_forward_selection(X, y, k_max=13)` and the function `ErrorEstimate(x,y,S)` is shown as:

```
# Estimate the 5-fold cross-validation error of s
def error_estimate(X, y, selected_features):
    model = LogisticRegression(penalty=None, max_iter=1000)
    X_subset = X[selected_features]
    cv_error = 1 - np.mean(cross_val_score(model, X_subset, y, cv=5,
                                           scoring='accuracy')) # k=5
    return cv_error

# Impletement stepwise forward algorithm
def stepwise_forward_selection(X_train, y_train, X_test, y_test, k_max
                                          =13):
    selected_features = []
    features = list(X_train.columns)
    cv_errors = []

for k in range(k_max): # Repeat k times
    best_error = float('inf')
    best_feature = None
    for f in features:
        if f not in selected_features:
            candidate_features = selected_features + [f] # features: S
                                                     U {f}
            err = error_estimate(X_test, y_test, candidate_features)
            if err < best_error:
                best_error = err
                best_feature = f
    selected_features.append(best_feature)
    cv_errors.append(best_error)
return selected_features, cv_errors
```

In this way, we can get a list of selected features, which are in order based on the cross validation error on the test data set.

(b) The plot showing the test error and cross-validation error for all values of k is shown as Fig. 2:

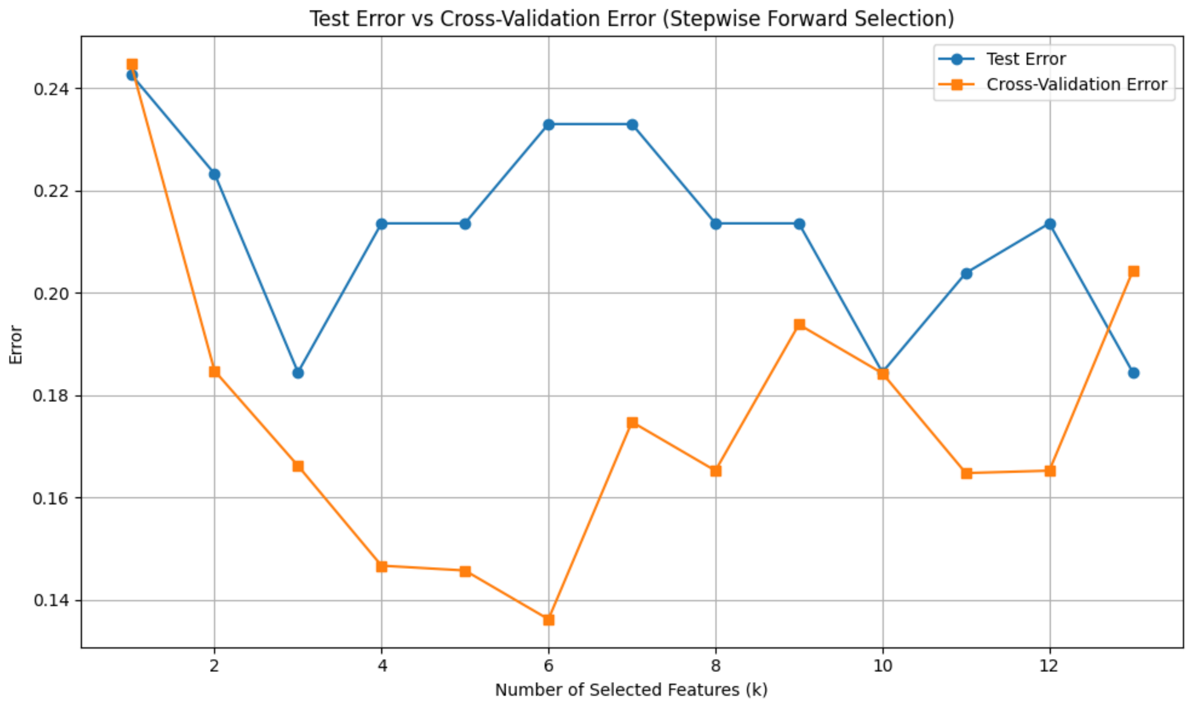

Figure 2: Problem 7(b): The test error and cross-validation error for k=1,2,...,13

(c) The two features that I select is: `cp` and `oldpeak`.
The decision boundary in this case is shown as Fig. 3:

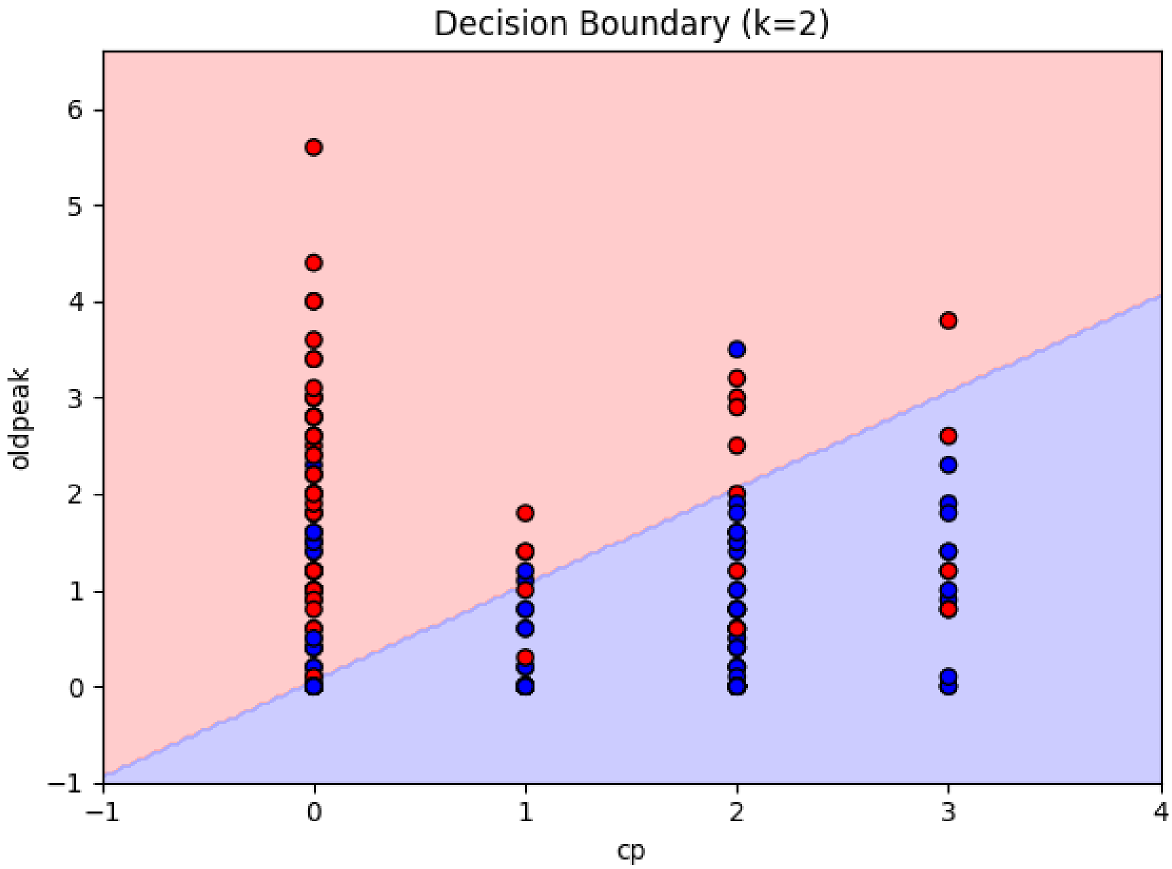

Figure 3: Problem 7(c): decision boundary of the model trained on selected features when k=2