Final Analytics Project (Individual Project)

Jhalak Sadana

INTRODUCTION

B-cells inducing antigen-specific immune responses in vivo produce large amounts of antigen-specific antibodies by recognizing the subregions (epitope regions) of antigen proteins. They can inhibit their functioning by binding antibodies to antigen proteins. Predicting of epitope regions is beneficial for the design and development of vaccines aimed to induce antigen-specific antibody production.

For this purpose of report, we are using the B-cell epitope data. In this we will use the information of protein and peptide to prediction whether or not an amino acid peptide exhibited antibody-inducing activity.

For the prediction purpose, our Y variable is binary (0 & 1). We will use Logistic regression for the model prediction and accessing it stability and accuracy.

DATA DESCRIPTION

- input_bcell.csv : this is our main data. The number of rows is 14387 for all combinations of 14362 peptides and 757 proteins.

The datasets consist of information of protein and peptide:

- parent_protein_id : parent protein ID
- protein_seq : parent protein sequence
- start_position : start position of peptide
- end_position : end position of peptide
- peptide_seq : peptide sequence
- chou_fasman : peptide feature, β turn
- emini : peptide feature, relative surface accessibility
- kolaskar_tongaonkar : peptide feature, antigenicity
- parker : peptide feature, hydrophobicity
- isoelectric_point : protein feature
- aromacity: protein feature
- hydrophobicity: protein feature
- stability: protein feature
- target : antibody valence (target value)

Overall description:

- 'parent_protein_id', 'protein_seq', and 'peptide_seq' are of object type as they contain characters not numbers.
- Rest of the features are of float type.
- No categorical feature is present in the dataset.
- Target feature is binary i.e. containing only 0 and 1.

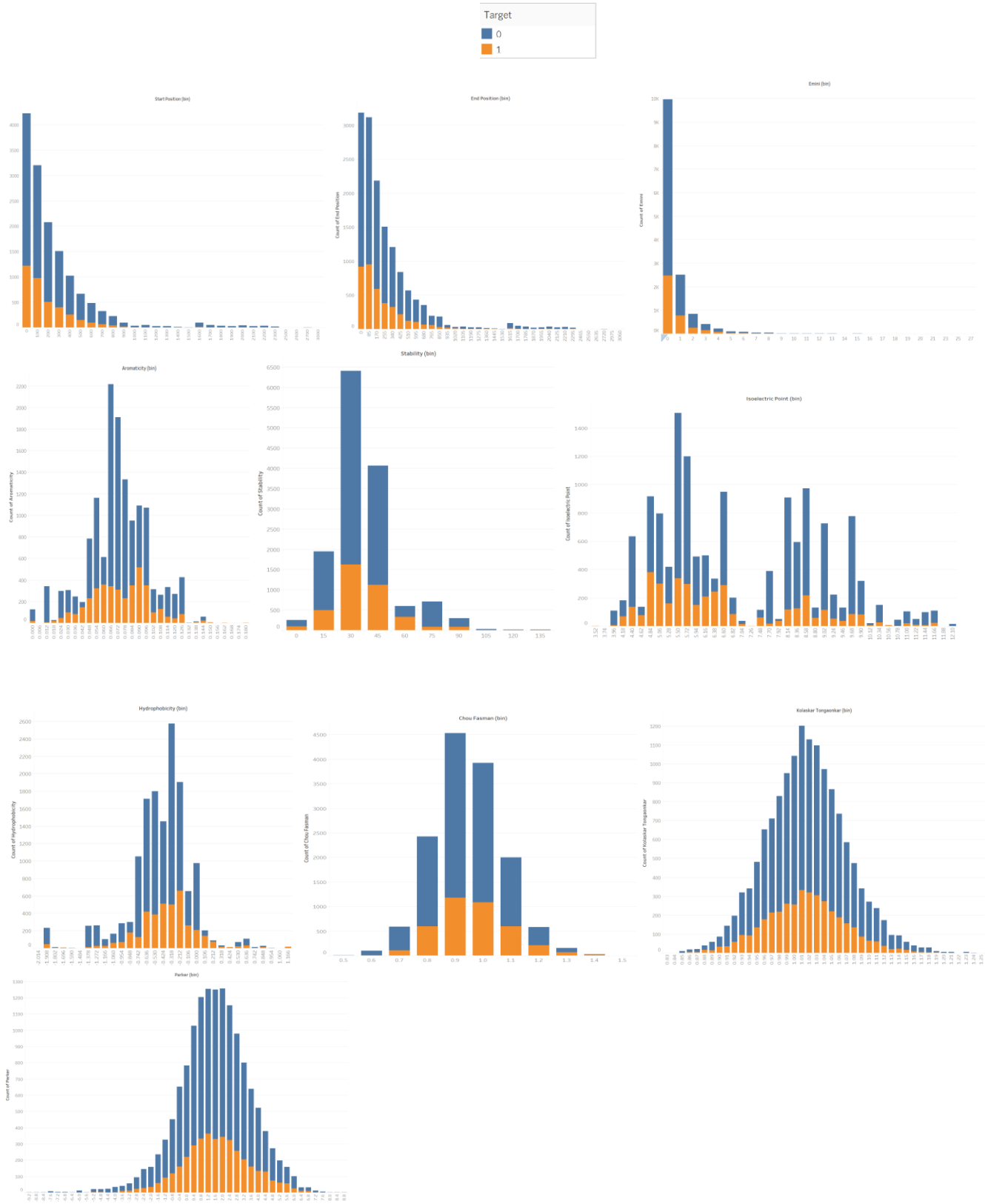
PROCESS:

A) Exploratory Data Analysis

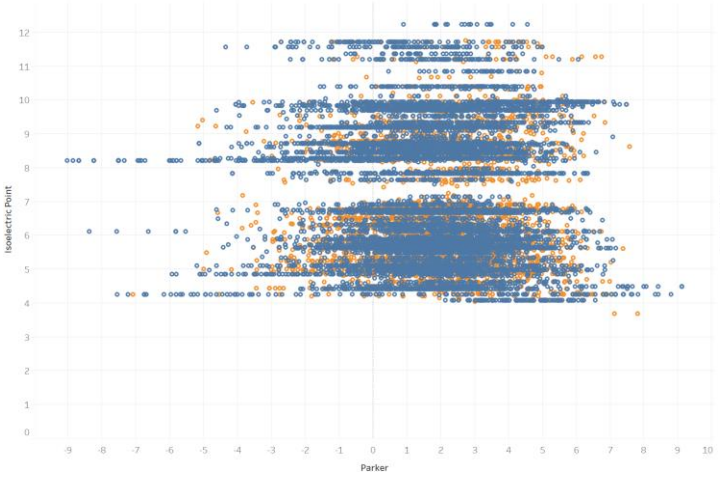
1.1. Checking missing values

```
> colSums(is.na(df))# shows the number of missing values in each column of the dataset
     start_position        end_position         chou_fasman               emini kolaskar_tongaonkar              parker 
                  0                   0                   0                   0                   0                   0 
  isoelectric_point        aromaticity      hydrophobicity           stability              target 
                  0                   0                   0                   0                   0 
```

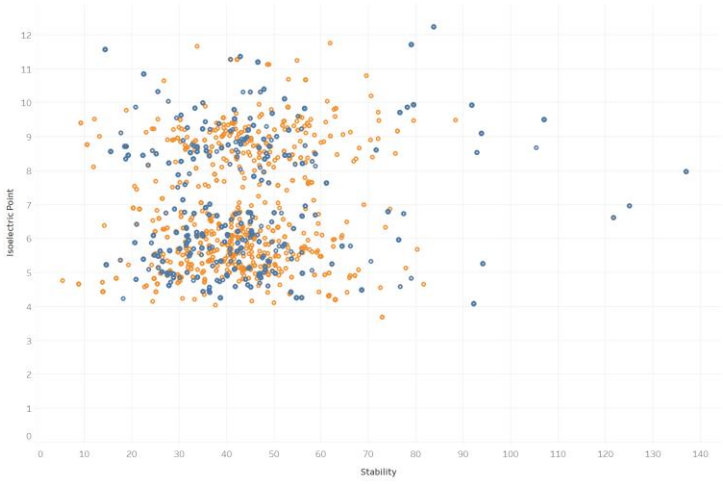
There are no missing values in the dataset.



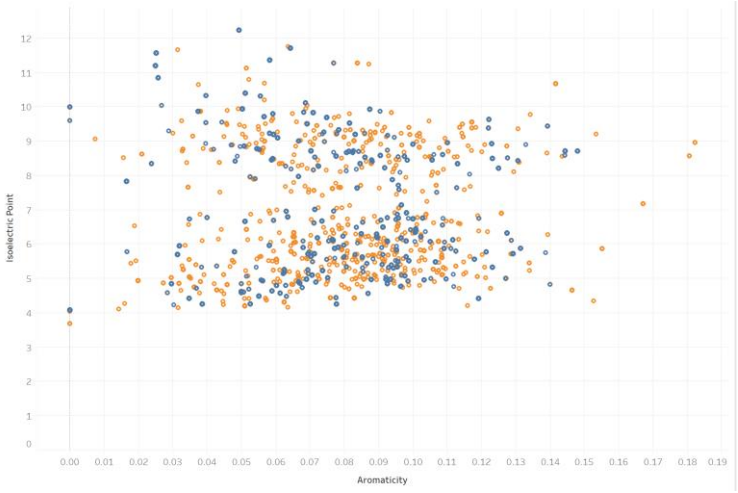
- 'Emini' feature shows right skewed distribution.
- 'peptide_length','isoelectric_point','aromaticity','hydrophobicity','stability' are not perfectly normal and contains outliers.
- 'chou_fasman','kolaskar_tongaonkar','parker' shows near-to-perfect normal distribution.



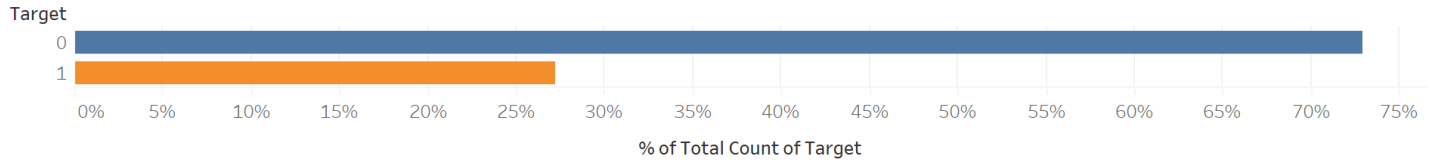
- Protein features 'stability','chiu fasman',"parker", "hydrophobicity" and 'isoelectric_point' conveys most information about the target feature determining their importance in the dataset.



- Most of the peptides showing strong hydrophobicity property are at positive target than negative target.
- Most of the peptides show stability within range 20 to 60 at positive target.
- Most of the peptides show stability within range 20 to 80 at negative target



- Most of the peptides have range of **aromaticity** within range of 0.05 to 0.10 at negative target.
- Most of the peptides have range of **aromaticity** within range of 0.04 to 0.12 at positive target.



72.8% target values i.e. Anitbody valence of this dataset is negative and 27.1% of values are positive. It means that most of the antibodies can resist binding of virus with themselves. So, we will use this original dataset without any changes. We also tried doing oversampling, but it didn't impact the accuracy of the model.

Additional EDA for the accuracy of the model

- We will remove the parent_protein_id, protein_seq, peptide_seq as it doesn't make sense to include in model in terms of its relation to the target value.
- We tried removing the lesser significant variables from the dataset for the model but it ultimately reduced the model accuracy and hence we kept it the original dataset with all the numeric variables to get the model with highest accuracy possible.

B) Divide the data into training and test

- Divided the B-cell dataset into train and test data.
- Train data contains 75% of data, which contains 10790 observations and 14 variables.
- Test data contains the remaining data, which contains 3597 observations and 14 variables.

C) Build the model on training set only

We are using logistic model. As the target variable is binary (0 &1). This logistic model is a statistical model that models the probability of an event taking place by having the log-odds for the event be a linear combination of one or more independent variables.

```
glm(formula = target ~ ., family = "binomial", data = df_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1057  -0.8251  -0.6560   1.2134   2.5432

Coefficients:
                   Estimate Std. Error z value             Pr(>|z|)
(Intercept)       -0.061180   0.685366  -0.089             0.928871
start_position    -0.018683   0.007991  -2.338             0.019394 *
end_position       0.017910   0.007994   2.240             0.025068 *
chou_fasman        2.011042   0.229054   8.780 < 0.0000000000000002 ***
emini              0.059263   0.014926   3.971            0.0000717 ***
kolaskar_tongaonkar -1.915766   0.597826  -3.205             0.001353 **
parker            -0.072273   0.018965  -3.811             0.000138 ***
isoelectric_point -0.204209   0.013119 -15.566 < 0.0000000000000002 ***
aromaticity        2.064148   0.992605   2.080             0.037569 *
hydrophobicity     0.715412   0.071352  10.027 < 0.0000000000000002 ***
stability          0.014729   0.001511   9.746 < 0.0000000000000002 ***
```

This trained model shows that all the variables are significant (* in all p values of coefficient).

D) Make predictions on test set.

- Using the predict () function R. we predict the Probability of an amino acid peptide exhibited antibody-inducing activity.
- This data is then classified into 0 and 1 based on the probability. If the probability is greater than 0.5, then it is classified as 1 and 0 in other case.
- Doing the same for the test and sample data prediction classification.

E) Create error distributions

Confusion matrix for training dataset and test dataset.

```
Confusion Matrix and Statistics

   train_pred
       0    1
 0 7713  141
 1 2736  200

               Accuracy : 0.7334
                 95% CI : (0.7249, 0.7417)
    No Information Rate : 0.9684
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0694

 Mcnemar's Test P-Value : <0.0000000000000002
```

```
Confusion Matrix and Statistics

   test_pred
       0    1
 0 2581   50
 1  917   49

               Accuracy : 0.7312
                 95% CI : (0.7164, 0.7456)
    No Information Rate : 0.9725
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0443

 Mcnemar's Test P-Value : <0.0000000000000002
```

Results :

```
> prop.table(table(df_train$target, train_pred))
   train_pred
             0          1
  0 0.71482854 0.01306766
  1 0.25356812 0.01853568
> prop.table(table(df_test$target, test_pred))
   test_pred
             0          1
  0 0.71754240 0.01390047
  1 0.25493467 0.01362246
```

Stability

Here we can see that the training dataset and test dataset has the true negatives as 71.48% and 71.75% respectively. In addition to that, the true positives are 1.18% and 1.13%, False negative are 1.3% and 1.39% and False positives are 25.35% and 25.49% respectively. All the values of test and training dataset are almost similar. This shows that are model is stable and robust. Its shows the similar prediction errors and correct prediction for both the dataset that it has seen (training dataset) and it has never seen i.e. test dataset.

Prediction Accuracy

The most basic diagnostic of a logistic regression is predictive accuracy. To understand this we need to look at confusion matrix proportion table displayed in above screenshot. The table above shows the prediction-accuracy table produced by logistic regression.

We will check upon the accuracy of the model that it has never seen which is our test predictions. In this we can see that the true positives and true negatives are 71.75% and 1.36% respectively. This sums up to 73.12% of accuracy. The model accurately predicts the amino acid peptide exhibited antibody-inducing activity for 73.12% of the test data. This tells us that for the 3597 observations used in the model for prediction, the model correctly predicted whether or not the amino acid peptide exhibited antibody-inducing activity in Bcell for 73.12% of the time. The accuracy of 73.12% is moderately fine as there is still good amount of data that is incorrectly predicted by the model.