CSM146: Homework 1

Alex Gilbert
904-770-190
Discussion 1D

2.) Total Entropy: $H(S) = B\left(\frac{p}{p+n}\right)$

Expected Entropy: (w/ x_j split) $\sum_{i=1}^{k} \frac{p_i+n_i}{p+n} B\left(\frac{p_i}{p_i+n_i}\right)$

$$= \sum_{i=1}^{k} \frac{p_i+n_i}{p+n} B\left(\frac{p}{p+n}\right)$$

$$= \frac{B\left(\frac{p}{p+n}\right)}{p+n} \sum_{i=1}^{k} (p_i+n_i) = \frac{B\left(\frac{p}{p+n}\right)}{p+n} [p+n]$$

$$= B\left(\frac{p}{p+n}\right)$$

$\therefore$ Information Gain = Total Entropy - Expected Entropy

$$= B\left(\frac{p}{p+n}\right) - B\left(\frac{p}{p+n}\right)$$

$$= 0 \checkmark$$

CSM146: Homework 1

Alex Gilbert
904-770-1910
Discussion 1D

---

3. a- Training set error is minimized for this dataset when $k=0$, for which training error $= 0$.

↳ Consider a set of examples & their labels (X, y) used to train a KNN classifier. Train error means finding y_{pred} for each X by plugging X into the model. If $k=0$, then for each $x \in X$, the model will set $y_{pred,x} = y_x$. Then, the train error will be $y_{pred} - y = 0 \ \forall \ x \in X$.

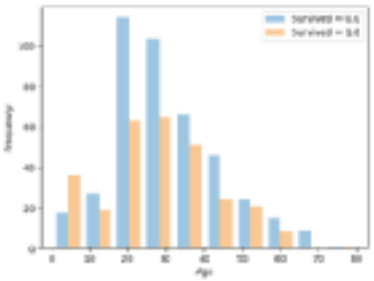
b- Too large values of k might be bad, since it would cause the model to consider training examples that aren't really related to a test example when determining the label for a test example.

Too small values of k might be bad, since the models predictions would suffer from outliers in the training examples. If $\hat{x}$ is a test example & its nearest neighbor is x', we might assign $\hat{y}_{pred} = y'$. If x' is an outlier & not really similar to $\hat{x}$: $\hat{y} \neq \hat{y}_{pred}$, we would get an error which might not occur if we considered nearby neighbors truly similar to $\hat{x}$.

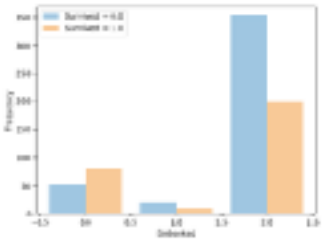
c- Best results occur for $k=5$ & $k=7$.

Using these k values, results in only 4 leave-one-out cross validation errors, out of 14 tests.

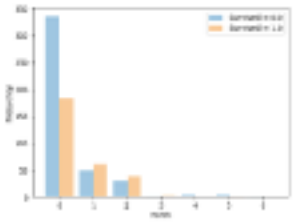
4.1.a) Feature Histograms…



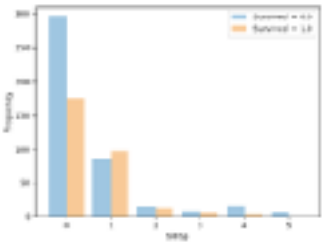
- **Age**: For every age group, only for the youngest infants and the most elderly, did fewer passengers die than survive. While young adults (20-30) were the most common passengers on board, they (along with ~40 year olds) had the smallest fraction of surviving passengers (roughly 40% or 30%). Towards the more extreme edges of the age range, the better your relative chances of survival.



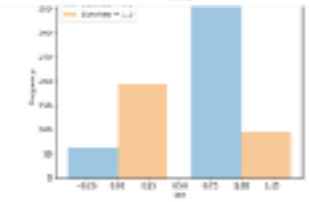
- **City of Embark**: Those who got on at Cherbourg appear to have had the best relative chance of survival (more survived the didn't), while only about 30% of passengers who boarded at the other two ports made it out alive. The majority of passengers boarded at Southampton, and very few at Queenstown.



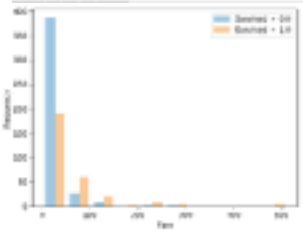
- **# of Parents/Kids on Board**: There were too few samples to make a statement about those with 3 or more Parents/kids members, but apparently having at least one parent or child greatly increased chances of survival, as more than 50% survived with 1 or 2 family members, and only about 30% of those traveling alone made it.



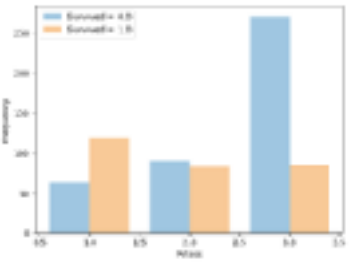
- **# of Siblings/Spouses**: Similar to parents/kids, having one sibling or spouse appears to have increased survival chances, but more than that seemed to have negative effects (though fewer samples present)



- **Sex**: Females had a much higher relative chance of survival than males.



- **Fare**: Paying more for your ticket seems to have improved chances of survival, but difficult to tell since so many more cheap tickets than expensive.



- **Ticket Class**: Survival chances decreased with increasing ticket class.

4.2.b) Random Classifier…

Achieved error of 0.485

```
Classifying using Random...
        -- training error: 0.485
```

4.2.c) Decision Tree Classifier…

Achieved training error of 0.014

```
Classifying using Decision Tree...
        -- training error: 0.014
```

4.2.d) KNeighbors Classifier…

K = 3:
    Achieved training error 0.167
K = 5:
    Achieved training error 0.201
K = 7:
    Achieved training error 0.240

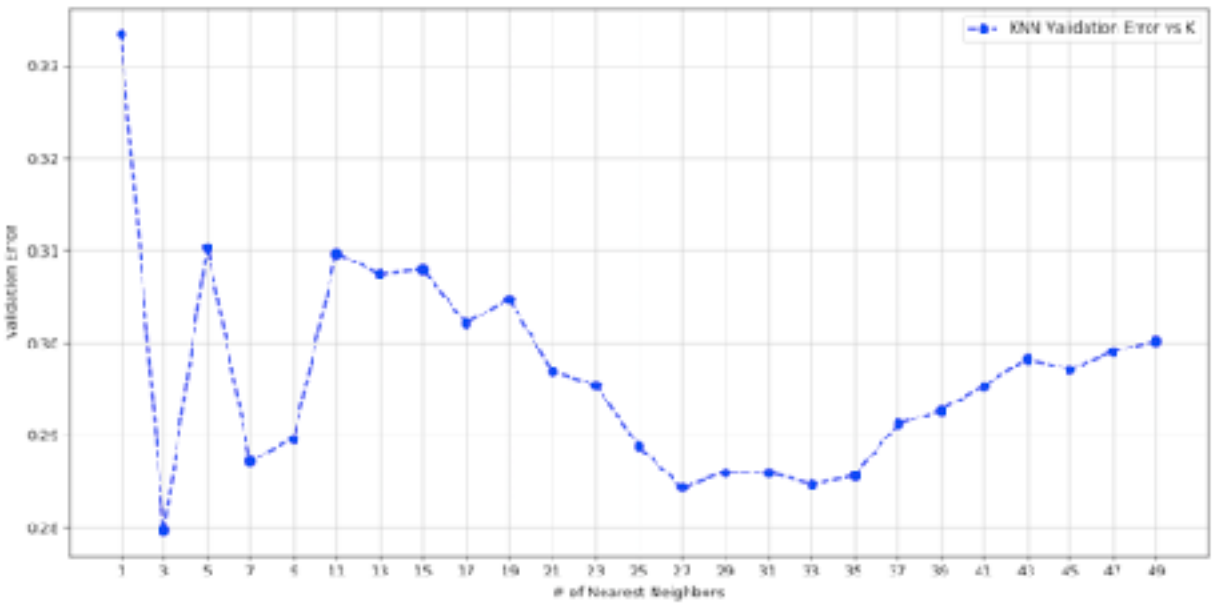
```
Classifying using 3 k-Nearest Neighbors...
        -- training error: 0.167
Classifying using 5 k-Nearest Neighbors...
        -- training error: 0.201
Classifying using 7 k-Nearest Neighbors...
        -- training error: 0.240
```

4.2.e) Cross Validation Error…

Majority vote:
    Avg Training error = 0.404
    Avg Test error = 0.407
Random:
    Avg Training error = 0.489
    Avg Test error = 0.487
Decision Tree:
    Avg Training error = 0.012
    Avg Test error = 241
KNeighbors:
    Avg Training error = 0.212
    Avg Test error =0.315

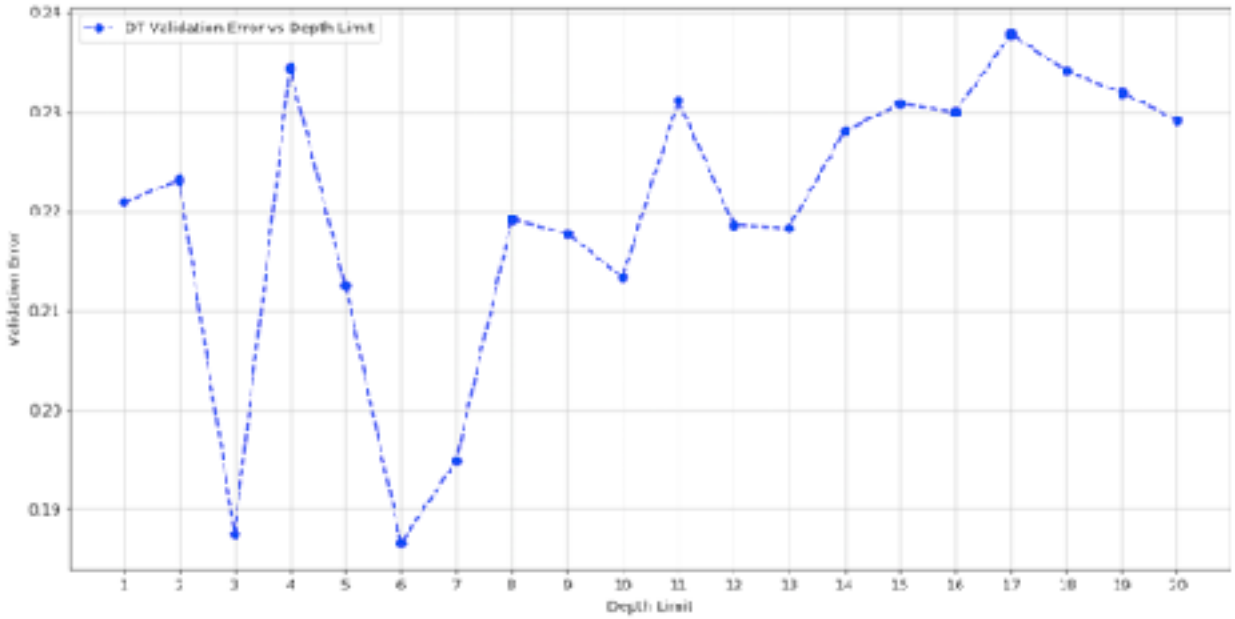
```
Investigating various classifiers...
MajorityVote: -- training error: 0.404  -- testing error: 0.407
Random: -- training error: 0.489        -- testing error: 0.487
DecisionTree: -- training error: 0.012  -- testing error: 0.241
K-Nearest: -- training error: 0.212     -- testing error: 0.315
```

4.2.f) 10-fold Cross Validation, Finding Best K…



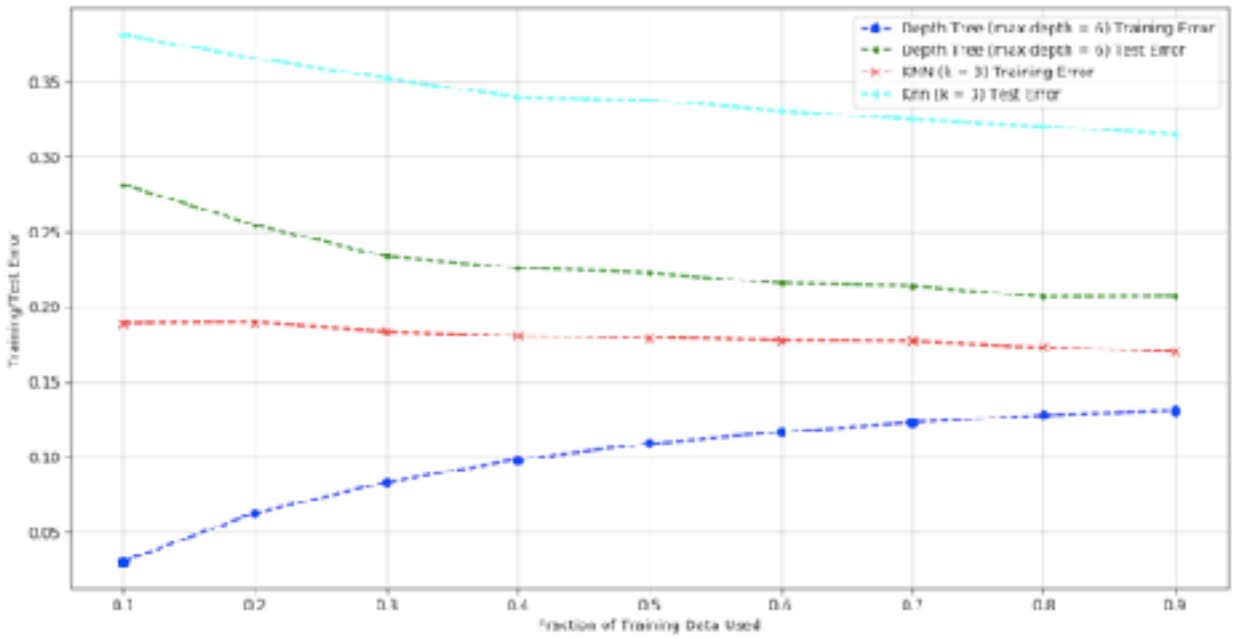
The **best value of K** (with the lowest cross validation error) is **K = 3.** Increasing K appears to have oscillating effects as you increment by 2…greatly improving from 1 to 3, then worsening again for 5, then gradually improving up to about 31, before worsening again approaching 49. In general, for small K, 3 works best, and for larger K (>10), (an odd number) around 30 is best.

4.2.g) 10-fold Cross Validation, Finding Best DT Depth Limit…



The **best max depth limit** (with the lowest cross validation error) is **d = 6.** Initially, increasing depth limit improves error rate, but past the optimal depth of 6, allowing the tree to go deeper progressively induces more cross-validation error. This is likely due to overfitting to the training data.

4.2.h) DT & KNN Test/Dev Learning Curves…



Overall test error is greater than training error as expected. The KNN classifier also outperforms the DT. Increasing the amount of training data used actually worsens the DT training error, likely due to overfitting. On the other hand, it always improves test error (though very slightly).