SENG 474

Assignment 1

Zhengtang Wang

V00802086

Q1.1

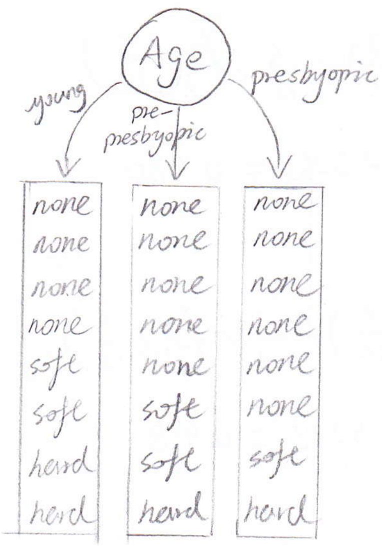
Age : $P(\text{young}) = \frac{8}{24}$, $P(\text{pre-presbyopic}) = \frac{8}{24}$, $P(\text{presbyopic}) = \frac{8}{24}$.

Spectacle-prescrip : $P(\text{myope}) = \frac{12}{24}$, $P(\text{hypermetrope}) = \frac{12}{24}$.

Astigmatism : $P(\text{no}) = \frac{12}{24}$, $P(\text{yes}) = \frac{12}{24}$.

Tear-prod-rate : $P(\text{reduced}) = \frac{12}{24}$, $P(\text{normal}) = \frac{12}{24}$.

Contact-lenses : $P(\text{soft}) = \frac{5}{24}$, $P(\text{hard}) = \frac{4}{24}$, $P(\text{none}) = \frac{15}{24}$.



Age = young :

$$\text{Entropy}\left(\frac{4}{8}, \frac{2}{8}, \frac{2}{8}\right) = -\frac{4}{8}\times\log_2\left(\frac{4}{8}\right) - \frac{2}{8}\times\log_2\left(\frac{2}{8}\right) - \frac{2}{8}\times\log_2\left(\frac{2}{8}\right)$$

$$= 0.5 + 0.5 + 0.5 = 1.5$$

Age = pre-presbyopic :

$$\text{Entropy}\left(\frac{5}{8}, \frac{2}{8}, \frac{1}{8}\right) = -\frac{5}{8}\times\log_2\left(\frac{5}{8}\right) - \frac{2}{8}\times\log_2\left(\frac{2}{8}\right) - \frac{1}{8}\times\log_2\left(\frac{1}{8}\right)$$

$$= 0.4238 + 0.5 + 0.375 = 1.299$$

Age = presbyopic :

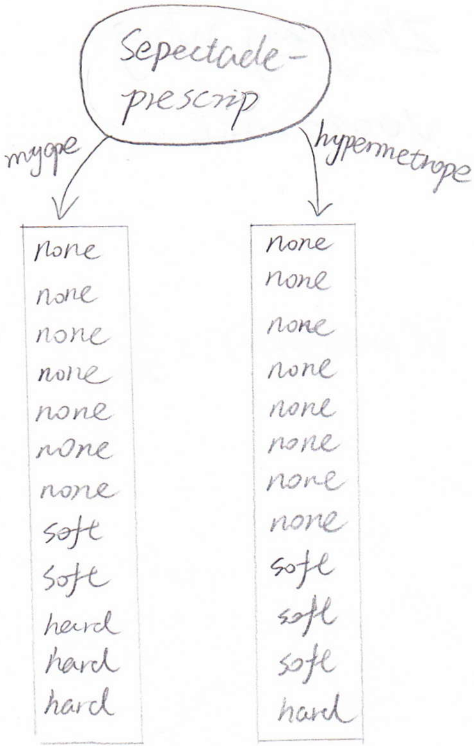
$$\text{Entropy}\left(\frac{6}{8}, \frac{1}{8}, \frac{1}{8}\right) = -\frac{6}{8}\times\log_2\left(\frac{6}{8}\right) - \frac{1}{8}\times\log_2\left(\frac{1}{8}\right) - \frac{1}{8}\times\log_2\left(\frac{1}{8}\right)$$

$$= 0.3113 + 0.375 + 0.375 = 1.061$$

Expected info :

$$AE = 1.5\times\left(\frac{8}{24}\right) + 1.299\times\left(\frac{8}{24}\right) + 1.061\times\left(\frac{8}{24}\right)$$

$$= 1.287$$



Sepectacle-prescrip = myope :

$$\text{Entropy}\left(\tfrac{7}{12}, \tfrac{2}{12}, \tfrac{3}{12}\right) = -\tfrac{7}{12}\times\log_2\left(\tfrac{7}{12}\right) - \tfrac{2}{12}\times\log_2\left(\tfrac{2}{12}\right) - \tfrac{3}{12}\times\log_2\left(\tfrac{3}{12}\right)$$

$$= 0.4536 + 0.4308 + 0.5 = 1.384$$

Sepectacle-prescrip = hypermetrope :

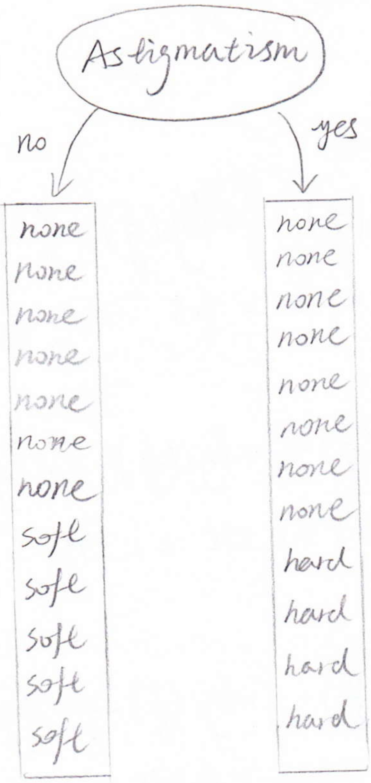
$$\text{Entropy}\left(\tfrac{8}{12}, \tfrac{3}{12}, \tfrac{1}{12}\right) = -\tfrac{8}{12}\times\log_2\left(\tfrac{8}{12}\right) - \tfrac{3}{12}\times\log_2\left(\tfrac{3}{12}\right) - \tfrac{1}{12}\times\log_2\left(\tfrac{1}{12}\right)$$

$$= 0.3900 + 0.5 + 0.2987 = 1.189$$

Expected info :

$$AE = 1.384\times\left(\tfrac{12}{24}\right) + 1.189\times\left(\tfrac{12}{24}\right)$$

$$= 1.287$$



Astigmatism = no :

$$\text{Entropy}\left(\tfrac{7}{12}, \tfrac{5}{12}\right) = -\tfrac{7}{12}\times\log_2\left(\tfrac{7}{12}\right) - \tfrac{5}{12}\times\log_2\left(\tfrac{5}{12}\right)$$

$$= 0.4536 + 0.5263 = 0.980$$

Astigmatism = yes :

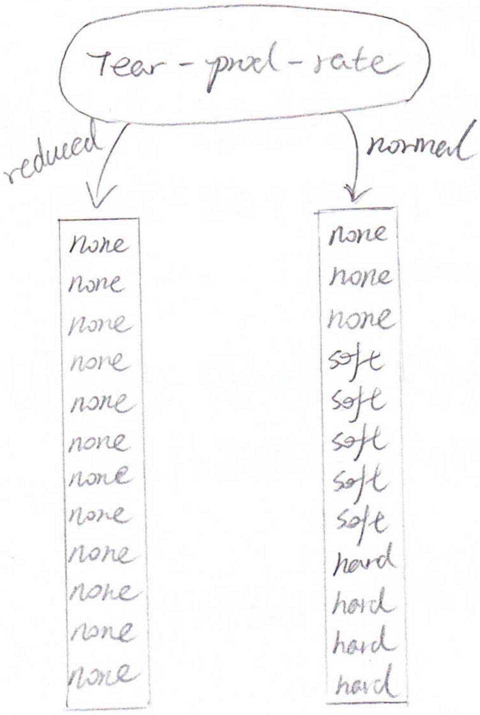
$$\text{Entropy}\left(\tfrac{8}{12}, \tfrac{4}{12}\right) = -\tfrac{8}{12}\times\log_2\left(\tfrac{8}{12}\right) - \tfrac{4}{12}\times\log_2\left(\tfrac{4}{12}\right)$$

$$= 0.3900 + 0.5283 = 0.918$$

Expected info :

$$AE = 0.98\times\left(\tfrac{12}{24}\right) + 0.918\times\left(\tfrac{12}{24}\right)$$

$$= 0.949$$



Tear - prod - rate = reduced:

$$\text{Entropy}\left(\frac{12}{12}\right) = -\frac{12}{12} \times \log_2\left(\frac{12}{12}\right)$$

$$= 0$$

Tear - prod - rate = normal:

$$\text{Entropy}\left(\frac{3}{12}, \frac{5}{12}, \frac{4}{12}\right) = -\frac{3}{12} \times \log_2\left(\frac{3}{12}\right) - \frac{5}{12} \times \log_2\left(\frac{5}{12}\right) - \frac{4}{12} \times \log_2\left(\frac{4}{12}\right)$$

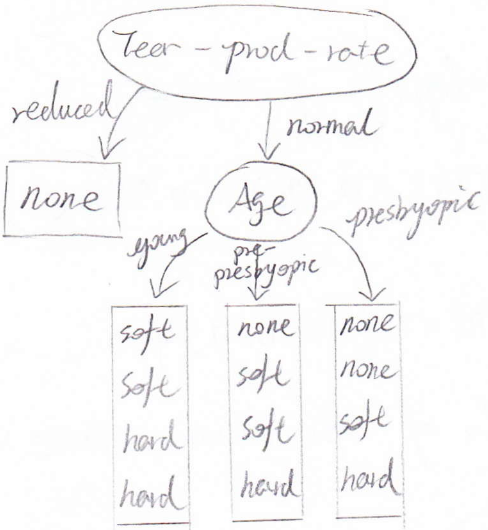
$$= 0.5 + 0.5263 + 0.5283 = 1.555$$

Expected info:

$$AE = 0 \times \left(\frac{12}{24}\right) + 1.555 \times \left(\frac{12}{24}\right) = 0.778$$

So, the root will be Tear - prod - rate .

Continuing to split.



Age = young:

$$\text{Entropy}\left(\frac{2}{4}, \frac{2}{4}\right) = -\frac{2}{4} \times \log_2\left(\frac{2}{4}\right) - \frac{2}{4} \times \log_2\left(\frac{2}{4}\right)$$

$$= 0.5 + 0.5 = 1$$

Age = pre - prebyopic:

$$\text{Entropy}\left(\frac{1}{4}, \frac{2}{4}, \frac{1}{4}\right) = -\frac{1}{4} \times \log_2\left(\frac{1}{4}\right) - \frac{2}{4} \times \log_2\left(\frac{2}{4}\right) - \frac{1}{4} \times \log_2\left(\frac{1}{4}\right)$$

$$= 0.5 + 0.5 + 0.5 = 1.5$$

Age = prebyopic:

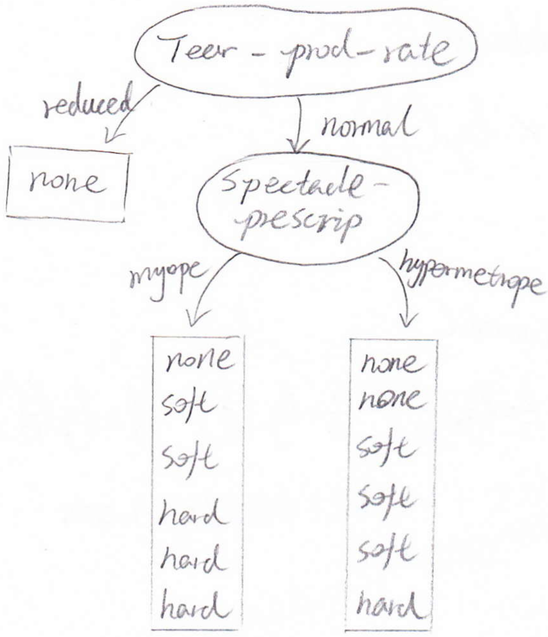
$$\text{Entropy}\left(\frac{2}{4}, \frac{1}{4}, \frac{1}{4}\right) = -\frac{2}{4} \times \log_2\left(\frac{2}{4}\right) - \frac{1}{4} \times \log_2\left(\frac{1}{4}\right) - \frac{1}{4} \times \log_2\left(\frac{1}{4}\right)$$

$$= 0.5 + 0.5 + 0.5 = 1.5$$

Expected info:

$$AE = 1 \times \left(\frac{4}{12}\right) + 1.5 \times \left(\frac{4}{12}\right) + 1.5 \times \left(\frac{4}{12}\right)$$

$$= 1.333$$



Spectacle - prescrip = myope :

$$\text{Entropy}\left(\frac{1}{6},\frac{2}{6},\frac{3}{6}\right) = -\frac{1}{6}\times\log_2\left(\frac{1}{6}\right) - \frac{2}{6}\times\log_2\left(\frac{2}{6}\right) - \frac{3}{6}\times\log_2\left(\frac{3}{6}\right)$$

$$= 0.4308 + 0.5283 + 0.5 = 1.459$$

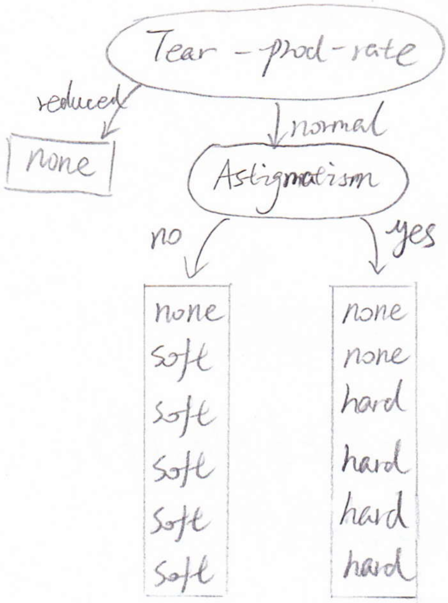
Spectacle - prescrip = hypermetrope :

$$\text{Entropy}\left(\frac{2}{6},\frac{3}{6},\frac{1}{6}\right) = -\frac{2}{6}\times\log_2\left(\frac{2}{6}\right) - \frac{3}{6}\times\log_2\left(\frac{3}{6}\right) - \frac{1}{6}\times\log_2\left(\frac{1}{6}\right)$$

$$= 0.5283 + 0.5 + 0.4308 = 1.459$$

Expected info :

$$AE = 1.459\times\left(\frac{6}{12}\right) + 1.459\times\left(\frac{6}{12}\right) = 1.459$$



Astigmatism = no :

$$\text{Entropy}\left(\frac{1}{6},\frac{5}{6}\right) = -\frac{1}{6}\times\log_2\left(\frac{1}{6}\right) - \frac{5}{6}\times\log_2\left(\frac{5}{6}\right)$$

$$= 0.4308 + 0.2192 = 0.650$$

Astigmatism = yes :

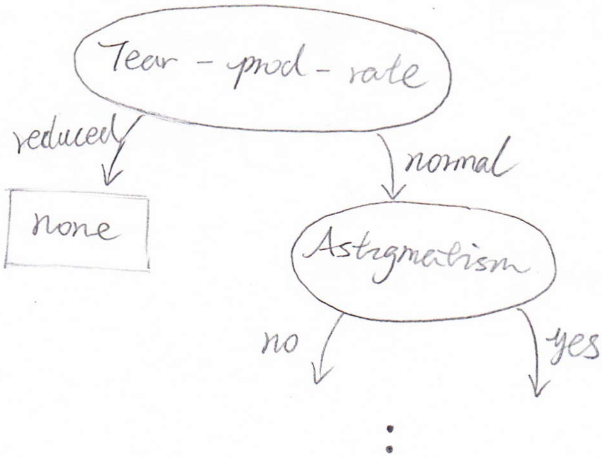
$$\text{Entropy}\left(\frac{2}{6},\frac{4}{6}\right) = -\frac{2}{6}\times\log_2\left(\frac{2}{6}\right) - \frac{4}{6}\times\log_2\left(\frac{4}{6}\right)$$

$$= 0.5283 + 0.3900 = 0.918$$

Expected info :

$$AE = 0.65\times\left(\frac{6}{12}\right) + 0.918\times\left(\frac{6}{12}\right) = 0.784$$

Therefore, the root and the first level of this decision tree will be :



⋮

Q1.2

According to the document for decision trees, scikit-learn uses an optimised version of the CART algorithm, which has a splitting method different from the ID3 algorithm, also calculated different entropies, then leads to the trees are not the same. CART uses Gini impurity.

Q2

$P(\text{none} \mid E)$

$= P(\text{prepresbyopic} \mid \text{none}) P(\text{hypermetrope} \mid \text{none}) P(\text{yes} \mid \text{none}) P(\text{reduced} \mid \text{none}) P(\text{none}) / P(E)$

$= \left(\frac{5+1}{15+3}\right)\left(\frac{8+1}{15+2}\right)\left(\frac{8+1}{15+2}\right)\left(\frac{12+1}{15+2}\right)\left(\frac{15+1}{24+3}\right) / P(E) = 0.0423 / P(E)$

$P(\text{soft} \mid E)$

$= P(\text{prepresbyopic} \mid \text{soft}) P(\text{hypermetrope} \mid \text{soft}) P(\text{yes} \mid \text{soft}) P(\text{reduced} \mid \text{soft}) P(\text{soft}) / P(E)$

$= \left(\frac{2+1}{5+3}\right)\left(\frac{3+1}{5+2}\right)\left(\frac{0+1}{5+2}\right)\left(\frac{0+1}{5+2}\right)\left(\frac{5+1}{24+3}\right) / P(E) = 0.000972 / P(E)$

$P(\text{hard} \mid E)$

$= P(\text{prepresbyopic} \mid \text{hard}) P(\text{hypermetrope} \mid \text{hard}) P(\text{yes} \mid \text{hard}) P(\text{reduced} \mid \text{hard}) P(\text{hard}) / P(E)$

$= \left(\frac{1+1}{4+3}\right)\left(\frac{1+1}{4+2}\right)\left(\frac{4+1}{4+2}\right)\left(\frac{0+1}{4+2}\right)\left(\frac{4+1}{24+3}\right) / P(E) = 0.00245 / P(E)$

Since $P(\text{none} \mid E) + P(\text{soft} \mid E) + (\text{hard} \mid E) = 1$

Then $\frac{0.0423}{P(E)} + \frac{0.000972}{P(E)} + \frac{0.00245}{P(E)} = 1$

$P(E) = 0.0423 + 0.000972 + 0.00245 = 0.0457$.

$P(\text{none} | E) = 0.0423 / P(E) = 0.0423 / 0.0457 = 0.926$

$P(\text{soft} | E) = 0.000972 / P(E) = 0.000972 / 0.0457 = 0.021$

$P(\text{hard} | E) = 0.00245 / P(E) = 0.00245 / 0.0457 = 0.054$

Therefore, 'prepresbyopic, hypermetrope, yes, reduced, ?' should be classified as 'none'. with the probability 0.926.

Q3.1

```
def train(self, X, y):
   # Your code goes here.
   # Calculate P(y) for each of the classes (y).
   cls, self._Ncls = np.unique(y, return_counts = True)
   self._Nfeat = len(X)
   self._class_prob = cls/len(y)
   # Calculate P(xi = 0|y) and for each y and every feature xi.
   self._Nfeat = np.zeros((len(self._Ncls), self._Nfeat))
   _, Nfeat = X.shape
   fc = np.zeros((len(self._Ncls), Nfeat))
   for i, count in enumerate(X):
     fc[y[i]] += count # Feature counts
   # Implement additive smoothing with α = 1.
   fc += self._smooth
   total = self._Ncls + (2 * self._smooth)
   # Probability of features.
   denominator = total.reshape(len(total), 1)
   self._feat_prob = fc/denominator
   return

def predict(self, X):
   # This is just a place holder so that the code still runs.
   # Your code goes here.
   pred = np.zeros(len(X))
   Xprob = np.ones(self._Ncls)
   for i in range(len(X)):
      s = X[i]
      for j in range(self._Ncls):
         Xprob[j] = self._class_prob[j]
         for f, feature in enumerate(s):
            if feature == 0:
               Xprob[j] *= (1 - self._feat_prob[j][f])
              else:
                 Xprob[j] *= (self._feat_prob[j][f])
      pred[i] = cls[Xprob.argmax()]
   return pred
```