Feature Subset Selection using Clusters & Informed Search
The Problem

1. Edge features
   - (a)
   - (b)
   - (c)
   - (d)

2. Line features
   - (a)
   - (b)
   - (c)
   - (d)
   - (e)
   - (f)
   - (g)
   - (h)

3. Center-surround features
   - (a)
   - (b)

Here I will be discussing exactly what the problem is (classification based on features of orientation) and what other types of problems our Solution aims to solve. Specifically, our algorithms are applicable to any normalized, continuous Feature sets.
I also plan to mention the (obvious) goal that we aim to minimize the number of features we're using for the sake of time and space complexity. If you want me to make another visualization for the masses (that is, something dumb and relatable to get people looking at our Slides before we hit the important stuff) I can.
OUTLINING OUR PROCESS

- What we learned from Phase 1:
  - Implementing tree/graph search over 1089 attributes is not an efficient solution!
  - Although the exhaustive search will certainly generate an optimal solution, it would take too long. This small subset would take days to search over using modern computers, and larger sets increase time complexity exponentially.

- Our approach to Phase 2- Reducing the dimensionality of search space:
  - grouping similar attributes into clusters
  - We did not use tree/graph search but we used greedy search to pick the best clusters
**Sampling Strategy 1**

- Get F1 score of each attribute individually
- Arrange the F1 scores and attributes in a decreasing manner

<table>
<thead>
<tr>
<th>Attribute Number</th>
<th>F1 Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>55</td>
<td>90%</td>
</tr>
<tr>
<td>56</td>
<td>78%</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>1089</td>
<td>0%</td>
</tr>
</tbody>
</table>

- Pick 5 or 10 or 20 attributes with the lowest attribute scores and remove them
- Evaluate removal of these attributes on overall F1 score
**Clusters**

- Why use clusters:
  - Group attributes that vary similarly/dissimilarly in the data set
  - Reducing time complexity and dimensionality in the search space

- Different types of clusters:
  - **Similar clusters**: Contains attributes that are similar to each other. Remove attributes that affect final F1 score in a similar manner and can be removed in one move.
  - **Dissimilar clusters**: Attributes that complement each other – each feature contains information that the others in the cluster do not. Discriminating attributes that define the data set.
  - **Random clusters**: Used for baseline results. Should give us similar F1 score as to when 1089 features are used.
Creating clusters using K-Means

- What is the position of an attribute?
- How do we measure distance between attributes?
- We need a representation for attributes such that clusters have a meaningful interpretation
CREATING CLUSTERS

- Let \( x_k \) be the \( k \)th instance in the dataset \( D \)
- Let \( a_k \) be the vector of attributes for instance \( x_k \)
- How do the individual attributes vary over each instance in \( D \)?
**CREATING CLUSTERS**

- First we calculate the covariance matrix:

\[
\text{Cov}(i, j) = \frac{\sum_{k=1}^{m} a_k(i) a_k(j)}{m} - \left( \frac{\sum_{k=1}^{m} a_k(i)}{m} \right) \times \left( \frac{\sum_{k=1}^{m} a_k(j)}{m} \right)
\]

where:
- \( m = \text{number of instances} \)
- For \( i, j = 1 \ldots \text{number of attributes} \)

\[\text{E}_D[a(i) \times a(j)] \quad \text{E}_D[a(i)] \quad \text{E}_D[a(j)]\]
Creating Clusters

- Then using the covariance matrix, we calculate the correlation matrix:

\[ m = \text{number of instances} \]

For \( i, j = 1 \ldots \text{number of attributes} \)

\[ \text{Corr}(i, j) = \frac{\text{Cov}(i, j)}{\sqrt{\text{Cov}(i, i) \times \text{Cov}(j, j)}} \]

<table>
<thead>
<tr>
<th>Correlation Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a_1 )</td>
</tr>
<tr>
<td>( a_1 )</td>
</tr>
<tr>
<td>( a_2 )</td>
</tr>
<tr>
<td>( a_3 )</td>
</tr>
<tr>
<td>( \ldots )</td>
</tr>
<tr>
<td>( a_k )</td>
</tr>
</tbody>
</table>

Range of Values from -1 to 1:

-1: Attributes give contradictory information

1: Attributes are very similar to each other

0: Attributes are independent
HILL CLIMBING – FEATURE SET REMOVAL

Make new feature subset

Bad

Test Heuristic

Good

Save changes

All features

Bad

No

# iteration < MAX_Iteration

No

Yes

Yes

Halt

Is Goal?

No

Yes

End
How features were removed

Phase I
F1 Scores by feature

{ $F_1$: $f_1=0.2001$, $F_2$: $f_1=0.2003$, $F_3$: $f_1=0.2007$, ... }

Phase II
Created Priority Queue of features

Each iteration n removed $x =\{1, 5, 10, 20\}$ features
HILL CLIMBING – CLUSTERS

- Start

- All features

- Get Clusters

- Start Search

- Select a cluster randomly

- Test Heuristic Accept move?
  - Yes
    - Remove cluster
  - No
    - Evaluate another cluster

- Is Goal or Max iterations reached?
  - Yes
    - Return feature subset
    - End
  - No
    - All features
HOW CLUSTERS WERE REMOVED

Created pre-processed set of clusters

Each iteration n removed 1 cluster randomly

\{ F_1 : f1=0.2001, 
  F_2 : f1=0.2003, 
  F_3 : f1=0.2007, ... \}

\{ F_1 : f1=0.3001, 
  F_2 : f1=0.3053, 
  F_3 : f1=0.2057, ... \}

\{ F_1 : f1=0.1001, 
  F_2 : f1=0.9203, 
  F_3 : f1=0.2057, ... \}
SEARCH STRATEGY

- Hill climbing
- Completeness?
  - Always produces a feature set
- Optimal?
  - No guarantee best set
- Why use this strategy?
  - Need for speed!
HEURISTIC
HEURISTIC

**F1-measure**: measure of how close a subset is from the F1-measure of the goal subset

F1-measure heuristic is admissible because it never overestimates how far a feature subset is from the goal
ADMISSIBILITY

Transition model, say Remove:
Remove(subset\textsubscript{A}, feature) = \{subset\textsubscript{B}\}
with subset\textsubscript{B} = \{subset\textsubscript{A} − feature\}

Subset\textsubscript{A}
Features:\{…x, y, z\}

\textbf{Subset\textsubscript{B}}
Features:\{…x, , z\}

• \(\alpha = \text{subset}_{\text{Goal}}’s\ F1\text{-measure}\)
• \(\beta = \text{subset}_{A}’s\ F1\text{-measure}\)
• \(\pi = \text{subset}_{B}’s\ F1\text{-measure}\)

Remove\{Subset\textsubscript{A}, y\} Minimum-cost action
**CONSISTENCY**

\[ \Delta_2 < \Delta_1 \quad \Rightarrow \quad \Delta_1 = \Delta_{\text{Step}} + \Delta_2 \]
**Consistency**

\[
\Delta_2 > \Delta_1 \quad \implies \quad \Delta_1 < \Delta_{\text{Step}} + \Delta_2
\]

\[
\Delta_1 \leq \Delta_{\text{Step}} + \Delta_2 \quad \text{which satisfies the triangle inequality}
\]

\[\rightarrow \text{F1-measure Heuristic is Consistent and Admissible}\]
RESULTS
**Sampling Strategy 1**

- Removing 5, 10, 20 features based on individual F1 scores

*Figure 1:* F1 score, precision, recall values using sampling strategy 1 when sampling 5, 10 or 20 features. Results are from the final feature subset selected by the search algorithm in TestSet 3
**Sampling Strategy 1**

- Removing 5, 10, 20 features based on individual F1 scores

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>F1</th>
<th>Accuracy</th>
<th>FN</th>
<th>FP</th>
<th>TN</th>
<th>TP</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 features</td>
<td>0.7823</td>
<td>0.7807</td>
<td>72</td>
<td>17</td>
<td>157</td>
<td>160</td>
<td>0.903955</td>
<td>0.689655</td>
</tr>
<tr>
<td>10 features</td>
<td>0.7735</td>
<td>0.7807</td>
<td>76</td>
<td>13</td>
<td>165</td>
<td>152</td>
<td>0.921212</td>
<td>0.666667</td>
</tr>
<tr>
<td>20 features</td>
<td>0.7247</td>
<td>0.7512</td>
<td>84</td>
<td>17</td>
<td>172</td>
<td>133</td>
<td>0.886667</td>
<td>0.612903</td>
</tr>
</tbody>
</table>

Table 1: Performance statistics of sampling strategy 1
Figure 2: Compares different clustering methods with no clusters. The * show the highest precision and recall values obtained using clusters of dissimilar/complementary features.
**Clusters**

- Using different type of clusters:

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Test 3</th>
<th>Train</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dissimilar Features</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F1</td>
<td>0.7962</td>
<td>0.9305</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.6873</td>
<td>0.87</td>
</tr>
<tr>
<td>FN</td>
<td>656</td>
<td>13</td>
</tr>
<tr>
<td>FP</td>
<td>160</td>
<td>0</td>
</tr>
<tr>
<td>TN</td>
<td>1642</td>
<td>0</td>
</tr>
<tr>
<td>TP</td>
<td>1594</td>
<td>87</td>
</tr>
<tr>
<td>Precision</td>
<td>0.7084</td>
<td>1</td>
</tr>
<tr>
<td>Recall</td>
<td>0.9087</td>
<td>0.87</td>
</tr>
</tbody>
</table>

| **Similar Features** |        |       |
| F1             | 0.7835 | 0.9305 |
| Accuracy       | 0.7878 | 0.88  |
| FN             | 694    | 12    |
| FP             | 166    | 0     |
| TN             | 1636   | 0     |
| TP             | 1556   | 88    |
| Precision      | 0.6915 | 1     |
| Recall         | 0.9036 | 0.88  |

| **Random Clusters** |       |       |
| F1             | 0.7750 | 0.9362 |
| Accuracy       | 0.7813 | 0.88  |
| FN             | 724    | 12    |
| FP             | 162    | 0     |
| TN             | 1640   | 0     |
| TP             | 1526   | 88    |
| Precision      | 0.6782 | 1     |
| Recall         | 0.9040 | 0.88  |

| **No Clusters** |       |       |
| F1             | 0.7708 | 0.9247 |
| Accuracy       | 0.7783 | 0.86  |
| FN             | 740    | 14    |
| FP             | 158    | 0     |
| TN             | 1644   | 0     |
| TP             | 1510   | 86    |
| Precision      | 0.6711 | 1     |
| Recall         | 0.9052 | 0.86  |

Table 2: Summarizes performance statistics of clustering methods versus no clustering. The results describe the F1, Accuracy, Precision and Recall from features in clusters selected by our search strategy.
SUMMARIZING

- Clustering based on different attributes clearly produces better results ($F1=0.7962$) than the approaches we tried
- Randomly grouping attributes into $k$ number of clusters ($F1=0.775$)
- Using clusters of similar attributes ($F1=0.7835$)
- Using no clusters at all, i.e. all features ($F1=0.7708$)
  - Selecting features from clusters containing similar attributes produces similar results as sampling with 5 features in Sampling Strategy1 ($F1=0.7823$)
  - Current best feature subset contains 437 features
CONCLUSION
SIMILAR FEATURE SETS

{Car color, wheel color, background color}
{Camera orientation, car orientation}
{Car production year, car brand}
Dissimilar Features

{Car color, camera orientation, car production year}

{Wheel color, car orientation, car brand}
FUTURE STRATEGIES

SIMILAR CLUSTERS ARE REDUNDANT

DISSIMILAR CLUSTERS ARE NOISY
FUTURE STRATEGIES
A* VARIANT, USING CLUSTER SIZE AS g(X)