Managing Randomness to Enable Reproducible Machine Learning

Hana Ahmed, Jay Lofstead

June 3, 2022
About Me

1. Senior @ Scripps College
2. Math+Computer Science major, Writing minor
3. Year-Round Intern @ Sandia National Labs w/ Dr. Jay Lofstead
AGENDA

1. Randomness and reproducibility
2. Pseudo-random number generators
3. Experiments with randomness in ML
4. Conclusions and future works
Reproducibility in science and ML

Reproducibility:

“obtaining consistent results using the same input data, computational steps, methods, and code, and conditions of analysis” (National Information Standards Organization)

Why this is important for ML:

Ensuring that models can be regenerated

Understanding the variance in accuracy caused by randomness
Randomness in ML

Examples:

- Shuffling training data
- Randomized subsets of input features
- Random initial weights

*Pseudo-random number generators (PRNGs):* Algorithms that generate sequences of pseudo-random values.
Approach

- **3 ML algorithms**: Neural network, K-means, Naïve Bayes
- **6 datasets**: Heart Disease, Wine, Iris, Breast Tissue, Wisconsin Breast Cancer (WBC), Somerville Happiness
- **4 variables**: Random seed, Train/test ratio, Train data set, Test data set
- **PRNG**: mt19937 (Mersenne Twister)
Experiment 1

**Seed:** Varied
**Train/test ratio:** Fixed
**Train data:** Varied
**Test data:** Varied

- **NN:** 44.53% difference on Heart Disease, 45.17% difference on Wine.
- **K-Means:** 8.88% difference on Iris, 28.62% difference on Breast Tissue.
- **Naïve Bayes:** 5.64% difference on WBC, 17.3% difference on Somerville Happiness.
Experiment 2

Seed: **Varied**
Train/test ratio: **Fixed**
Train data: **Varied**
Test data: **Fixed**

- **NN:** 33.23% difference on Heart Disease, 0% difference on Wine.
- **K-Means:** 22.22% difference on Iris, 26.31% difference on Breast Tissue.
- **Naïve Bayes:** 0% difference on WBC, 0% difference on Somerville Happiness.
Experiment 3

Seed: **Fixed**
Train/test ratio: **Varied**
Train data set: **Varied**
Test data set: **Fixed**

Model performance varied widely for each algorithm on all data sets despite controlling the seed.

<table>
<thead>
<tr>
<th>Train/Test</th>
<th>Heart Disease (NN)</th>
<th>Wine (NN)</th>
<th>Iris (K-Means)</th>
<th>Breast Tissue (K-Means)</th>
<th>Wisconsin Breast Cancer (Naive Bayes)</th>
<th>Somerville Happiness (Naive Bayes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/90</td>
<td>0.4873</td>
<td>0</td>
<td>0.6296</td>
<td>0.2737</td>
<td>0.6455</td>
<td>0.5703</td>
</tr>
<tr>
<td>20/80</td>
<td>0.4873</td>
<td>0</td>
<td>0.5833</td>
<td>0.3095</td>
<td>0.6279</td>
<td>0.5701</td>
</tr>
<tr>
<td>30/70</td>
<td>0.7476</td>
<td>0.3296</td>
<td>0.523</td>
<td>0.4324</td>
<td>0.6155</td>
<td>0.56</td>
</tr>
<tr>
<td>40/60</td>
<td>0.4873</td>
<td>0.3296</td>
<td>0.5556</td>
<td>0.4127</td>
<td>0.5776</td>
<td>0.5294</td>
</tr>
<tr>
<td>50/50</td>
<td>0.4873</td>
<td>0.3296</td>
<td>0.6667</td>
<td>0.0566</td>
<td>0.5673</td>
<td>0.5211</td>
</tr>
<tr>
<td>60/40</td>
<td>0.5663</td>
<td>0.3296</td>
<td>0.8333</td>
<td>0.3571</td>
<td>0.5878</td>
<td>0.5263</td>
</tr>
<tr>
<td>70/30</td>
<td>0.5205</td>
<td>0.3296</td>
<td>1.0</td>
<td>0.4688</td>
<td>0.5789</td>
<td>0.5</td>
</tr>
<tr>
<td>80/20</td>
<td>0.8372</td>
<td>0.3296</td>
<td>1.0</td>
<td>0.3810</td>
<td>0.5971</td>
<td>0.4286</td>
</tr>
<tr>
<td>90/10</td>
<td>0.4873</td>
<td>0.3296</td>
<td>0</td>
<td>0.5455</td>
<td>0.7101</td>
<td>0.5</td>
</tr>
</tbody>
</table>
Conclusions

• Random number seeds significantly impact the quality of ML models.

• **Current work**: understanding the potential variance range for ML algorithms for any given data set

• **Future work**:
  ○ Experimenting with different types of ML algorithms and data sets.
  ○ Applying same techniques to parallel ML algorithms
  ○ Running identical experiments using a GPU
Thank you.

Contact: hahmed@sandia.gov