Experiment:
(Example) Pixel-Level Text Detection in Natural Scenes: Comparison of Nearest Neighbor and Linear Regression

(Your Name Here)

January 11, 2012

Purpose: This is a template for reporting experimental results not unlike that used in the traditional sciences. It's purpose is to provide a **brief** structured record of an experiment, and thereby a basis for analysis when planning work and writing papers from a series of experiments, and to assist discussions between members of the lab.

1 Hypothesis

Identify what the expected outcome for the experiment is. The hypothesis should be a concrete, testable statement regarding the experiment outcome.

For the running example, we might hypothesize that the nearest neighbor classifier will be more accurate, **in terms of f-measure**, than a linear regressor (simple neural net) trained using the same set of features used to classify individual image pixels as text or non-text.

Rationale: Here you should briefly explain why you believe your hypothesis to be true, and include any citations to important references (e.g. that identify key algorithms, theories, data sets, or evaluation techniques [1]). For the example, we might assume or have observed in earlier work or experiments that the features used are not linearly separable, making a linear regressor unlikely to separate pixel and non-pixel classes reliably (a simple linear regressor learns a linear separator for two classes).

The rationale should provided a brief but complete 'sketch' of the main ideas and sources that support your hypothesis; a complete discussion of the hypothesis and related work can be produced by expanding this sketch in the write-up of a research paper, if that turns out to be worthwhile.

2 Experimental Design

2.1 Scripts

Identify the scripts (e.g in Python, Bash, or your favorite scripting language of choice) that can be executed to run your experiment, and where they are located.

Example: `expNNLR.py`, a python script that runs the experiment below. This script should be designed so that you can **replicate (repeat) the experiment in the case where results get lost, a related experiment is designed, or an error is found.**

2.2 Data Set, Metrics, and Statistical Hypothesis Tests

Identify the data set to be used, **the directory or files that contain the data set, and where it can be found. Also identity features being used, providing a definition, or citing the paper(s) where features are defined**. Here this might be grey-level pixel intensity, latent values (i.e. the ratio of the coefficient of the first principal component over the sum of coefficients for all three PCs in a three-color space, considering the color values in pixels within a neighborhood of each pixel (e.g. a 5x5 grid with each pixel at the center)), and/or other features.

Identify every metric used in the experiment. When a metric is taken from a paper but not widely used, cite the paper and provide a reference for the paper defining the metric. Metrics like recognition rate, recall, precision, and f-measure (2*recall*precision)/(recall + precision) are widely used, and don't need to be described, just named. For the example, we would observe recall (percentage of text pixels found), precision (percentage of detected text pixels that are *actually* text), and f-measure (the harmonic mean of recall and precision, as defined above).

If appropriate, explicitly identify a hypothesis test (e.g. t-test, chi-square, wilcoxin ranked-sum, ANOVA) being used, along with the significance level (the 'p-value' at which the null hypothesis is considered as being rejected). **Increasingly claims about differences in the performance of algorithms must be supported by statistical hypothesis tests, particularly in high-impact conferences and journals.** For simplicity, our running example does not make use of a hypothesis test.

2.3 Algorithm(s)

This section identifies the different algorithm variations used in the experiment (i.e. the *experimental conditions*).

Global parameters: Any experimental parameters that are constant across each algorithm/condition should be identified.

1. Nearest Neighbor (NN)
   (a) Parameter set 1 (e.g. k = 3)
   (b) Parameter set 2 (e.g. k = 5)
2. Linear Regressor (single perceptron) (LR)
   (a) Parameter set 1 (e.g. epochs = 100, alpha (learning rate) = 0.1)
   (b) Parameter set 2 (e.g. epochs = 100, alpha (learning rate) = 0.05)

Implementations: You should make reference to the *specific* version of source code used in the experiment, e.g. `nnlr.cc` version 1.2. You should make use of SVN, CVS or another version control system so that you can easily recover versions of your code, without ever throwing a version away.

2.4 Additional Design Notes

In some cases, additional notes may be needed to understand the design of the experiment, e.g. if the experiment considers different feature sets as well as different algorithms.

3 Results

Show a table or plot (below we use a table) summarizing the experimental results. Plots often make it easier to see patterns (and spot errors) in results, so use them when it seems helpful, particularly when then is a lot of data, or when taking variance into account. For example, if you report the average of a metric, you should show error bars in a box plot (visualizing +/- 1 standard deviation) or use a box plot to give a sense of the distribution of errors, as many distributions have the same average; without being able to see the distribution, averages for a narrow range of values may be identical to those that vary dramatically across iterations. Also, bar graphs (with 'error bars') and box plots are commonly used to visualize data when using statistical hypothesis tests, such as t-tests, or ANOVA, and are generally expected in research publications when using hypothesis testing.

Produce the experimental data as one or more tables stored in files (e.g. in a text file, .mat MATLAB file, etc.) before producing graphics, allowing specific metric values to be found as needed, and easily imported later on into tools for analysis and visualization such as MATLAB, Octave, R, PyMat, etc.. Identify the name of this file, e.g. `results.txt`.

	Recall (%)	Precision (%)	F-measure (%)
NNa	20.0	80.0	32.0
NNb	50.0	50.0	50.0
LRa	70.0	70.0	70.0
LRb	45.0	95.0	61.1

Outcome: Indicate whether the result confirms or contradicts your hypothesis. Here, our hypothesis that the Nearest Neighbor algorithm would have better F-measure values is contradicted. Here LRa (with the larger learning rate) seems to learn the data best.

Additional Results: In many cases additional bar graphs, box plots, line graphs, or other visualizations used to visualize the results and analyze the data should be provided, to try and *better understand* the result of the experiment, as well as **catch any errors** in the algorithm implementations, experiment implementation, or design.

4 Discussion

Discuss how the results confirm or contradict the hypothesis. Consider possible causes, making reference to any additional results that have been collected that are pertinent. This section of the document may contain opinion; aside from the hypothesis, all other contents of the document should be factual.

For the example experiment, apparently treating the data as linearly separable is leading to better results than the nearest neighbor methods for k=3 and 5; in the discussion section we can make conjectures about why this is, ideally supported by the data analysis, but not necessarily. In our example, only a small range of parameter values were attempted for the algorithms; one might suggest that a 7-NN or 9-NN classifier might perform comparably or better than the linear regressors, though with additional computational cost.

The discussion should summarize the outcome of the experiment, and suggest any additional experiments that may be of interest.

References

[1] Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, The Google Books Team, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182, 2011.