Detecting Gerrymandering with Machine Learning Models

Final Dissertation Presentation

By Andrew Frisby

ID: 201525902

Advisor: Aris Filos-Ratsikas

Second Marker: Alkmini Sgouritsa

August 26, 2022

Agenda

- Project Description
- Aims and Objectives
- Outputs
- Evaluation
- Q&A

Project Overview

- Gerrymandering
  - Manipulation of voters by drawing voting districts that sway elections in favor of one party
  - Ex.)

	Party A	Party B
Popular Vote Share	49%	51%
Legislative Share	55%	45%

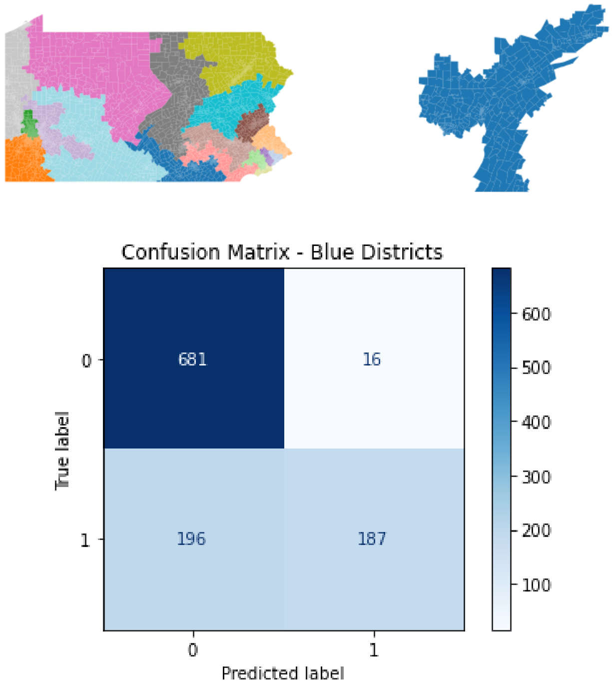
  - Packing and cracking
- Current methods measure it with statistics and outlier detection
  - There is an "absence of any 'limited and precise standard' for evaluating partisan gerrymandering" (*Rucho v. Common Cause*)
- Attempting to create a machine learning (ML) model to use as a standard for evaluating and detecting gerrymandering
  - A research-based, proof of concept dissertation using both image and numerical data with sophisticated ML models
  - Models combine both normal neural networks (NNs) and convolutional neural networks (CNNs)

Aims and Objectives

- Develop ML model to detect gerrymandering in a district and/or districting plan ☑
  - Generate realistic districting plans and accompanying demographic data ☑
  - Score and label generated plans ☑
  - Implement effective method to combine numerical demographic data and district images in single ML model ☑
  - Build CNN/NN/combined model ☑
  - Tune, train, and test model on labeled data ☑
- Discover most appropriate metrics and metric composites to label data used in model ☒
  - Incorporate and develop metrics to label data ☑
  - Compare results of models using different labeling metrics ☑
- Offer a standard for evaluating gerrymandering with machine learning ☑
- Illustrate power of gerrymandered districts on elections ☒
  - Visualize and report election results of most gerrymandered district plans and compare them to non-gerrymandered plans ☒

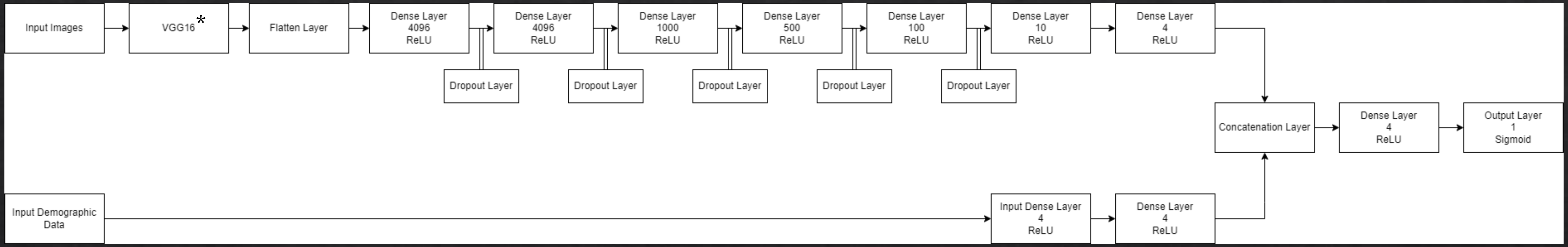
Outputs

- District and statewide images for each generated plan along with accompanying numerical demographic data
- Combined CNN and NN models that detect gerrymandering in districting plans for the state of Pennsylvania
- Visualizations of the results



	precision	recall	f1-score	support
False	0.78	0.98	0.87	697
True	0.92	0.49	0.64	383
accuracy			0.80	1080
macro avg	0.85	0.73	0.75	1080
weighted avg	0.83	0.80	0.78	1080

Outputs - “Basic” Model Architecture



* VGG16 architecture in backup slides

Evaluation

- Proved that machine learning models can reliably and accurately predict gerrymandering
  - Dependent on metrics used to define and label gerrymandering
    - Confident that models could produce sufficient results across a wide range of metrics
- Achieved relatively high accuracies that improved upon both baseline and initial models
  - Best models consistently reached 80-86% accuracy
  - Demographic data and initial VGG16 network proved to be sufficient
- Followed original design proposal and development cycle*
  - Intentionally vague to allow exploration of a wide range of research questions

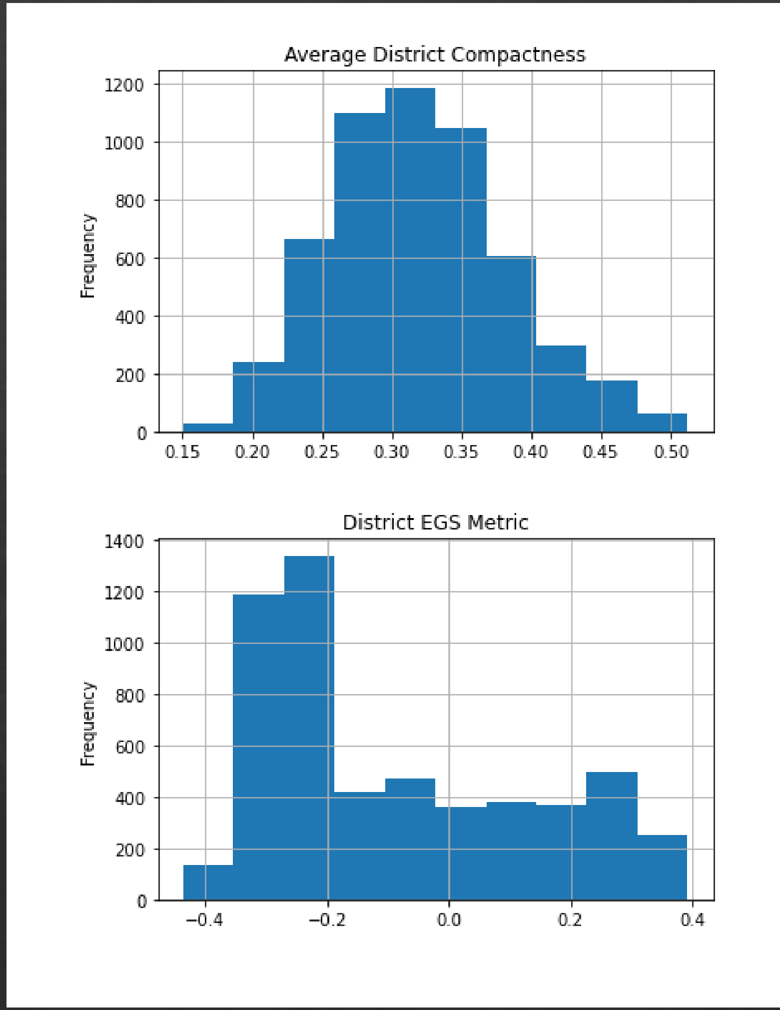
* Original development cycle in backup slides

Evaluation - Research Questions

- Can ML models detect/predict gerrymandering?
- How can one combine image and numerical data into a ML model?
- What model architecture(s) is/are most suitable for this prediction?
- Which demographic features should be included?
- How do the labeling metrics impact such models?
- Which metrics are the best for the ML models?
- Which input images perform better in the ML models, statewide or district?
- How to preprocess the images to get the best performance?
- What type of classification model is best? (Binary vs. multiclass)
- Does the starting districting plan impact the data generation and/or the model results?
- How do different district colors impact the model?
- Does oversampling to address the class imbalance impact the model?

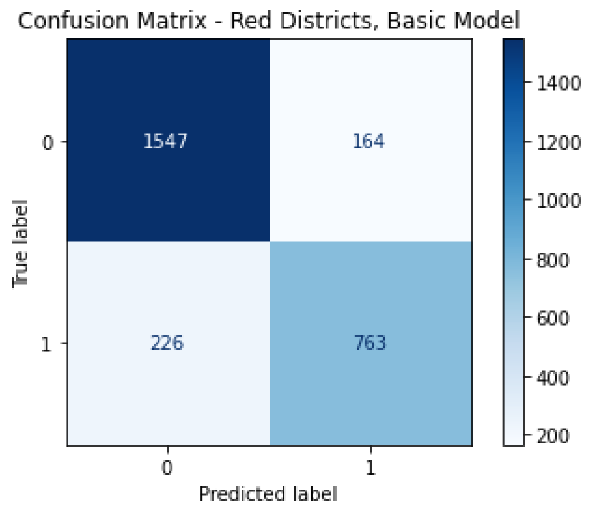
Evaluation – Changes and Shortfalls

- GerryChain library memory leak during data generation caused a reduction in planned total observations
- Balancing of classes required oversampling techniques which hindered the models’ performance and were ultimately deemed unnecessary and abandoned
- Keras Tuner was used for parameter searching/tuning
- Compactness metrics – majority of data <0.50, difficult to generate thresholds and labels
- Partisan metrics – most calculated scores on a statewide level making them obsolete for the district models
- Extremely long run times
- Intentions to examine other CNNs, different states, and implications of gerrymandering were not explored due to time constraints



Evaluation – Overall Results

- Initial model test accuracy: 65.8%
- Over 40 models generated and tested during development cycle*
- Keras-Tuner models typically only performed as good as "basic," untuned models
- Best model test accuracy: 86%
  - 13,500 observations
  - Red districts, but, overall, color determined not to be contributing factor across models
  - "Basic" model

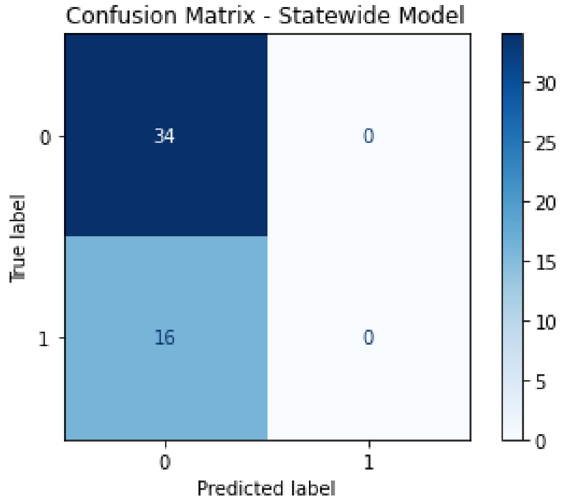


	precision	recall	f1-score	support
False	0.87	0.90	0.89	1711
True	0.82	0.77	0.80	989
accuracy			0.86	2700
macro avg	0.85	0.84	0.84	2700
weighted avg	0.85	0.86	0.85	2700

* More model results in backup slides

Evaluation – Statewide Results

- Initial model test accuracy: 65.8%
- Statewide model test accuracy: 68%
  - With oversampling of True class: 53.6%
- Impacted by data generation limitations and small sample size
  - Only 250 total observations



	precision	recall	f1-score	support
False	0.68	1.00	0.81	34
True	0.00	0.00	0.00	16
accuracy			0.68	50
macro avg	0.34	0.50	0.40	50
weighted avg	0.46	0.68	0.55	50

References

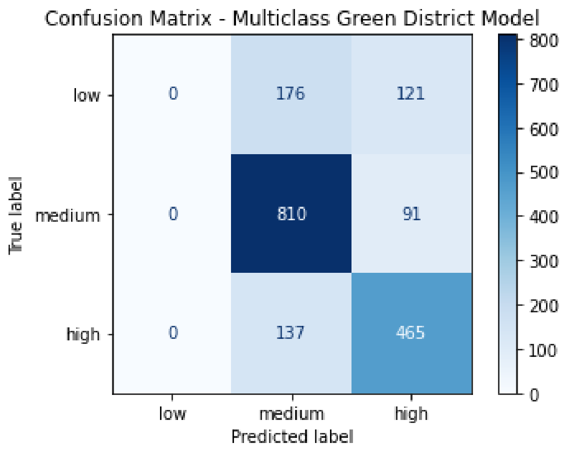
- Bangia, S. *et al.* (2015) *Quantifying Gerrymandering : Data+ 2015 @ Duke*. Available at: https://services.math.duke.edu/projects/gerrymandering/index-dataPlus.html (Accessed: 8 July 2022).
- Bangia, S. *et al.* (2017) 'Redistricting: Drawing the Line'. arXiv. Available at: http://arxiv.org/abs/1704.03360 (Accessed: 29 June 2022).
- Borodin, A. *et al.* (2018) 'Big City vs. the Great Outdoors: Voter Distribution and How It Affects Gerrymandering', in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. Twenty-Seventh International Joint Conference on Artificial Intelligence {IJCAI-18}*, Stockholm, Sweden: International Joint Conferences on Artificial Intelligence Organization, pp. 98–104. Available at: https://doi.org/10.24963/ijcai.2018/14.
- DeFord, D. (2019) 'How to build districting ensembles: A guide to GerryChain'.
- Fisher, Z. (2017) *Measuring Compactness*. Available at: https://fisherzachary.github.io/public/r-output.html (Accessed: 29 June 2022).
- 'GerryChain' (2022). Metric Geometry and Gerrymandering Group. Available at: https://github.com/mggg/GerryChain (Accessed: 29 June 2022).
- ul Hassan, M. (2018) 'VGG16 - Convolutional Network for Classification and Detection', 20 November. Available at: https://neurohive.io/en/popular-networks/vgg16/ (Accessed: 30 June 2022).
- Herschlag, G., Ravier, R. and Mattingly, J.C. (2017) 'Evaluating Partisan Gerrymandering in Wisconsin'. arXiv. Available at: http://arxiv.org/abs/1709.01596 (Accessed: 29 June 2022).
- 'Keras: Multiple Inputs and Mixed Data' (2019) *PyImageSearch*, 4 February. Available at: https://www.pyimagesearch.com/2019/02/04/keras-multiple-inputs-and-mixed-data/ (Accessed: 18 July 2022).
- Lewenberg, Y., Lev, O. and Rosenschein, J.S. (2017) 'Divide and Conquer: Using Geographic Manipulation to Win District-Based Elections', p. 9.
- Li, L. *et al.* (2018) 'Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization', p. 52.
- *Metric Geometry and Gerrymandering Group* (no date). Available at: https://github.com/mggg (Accessed: 29 June 2022).
- *MGGG States* (no date) *GitHub*. Available at: https://github.com/mggg-states (Accessed: 4 July 2022).
- *Redistricting Report Card Methodology* (2021). Available at: https://gerrymander.princeton.edu/redistricting-report-card-methodology (Accessed: 29 June 2022).
- Robert, C.P. (2016) 'The Metropolis-Hastings algorithm'. arXiv. Available at: http://arxiv.org/abs/1504.01896 (Accessed: 29 June 2022).
- *Rucho v. Common Cause* (2019) *Oyez*. Available at: https://www.oyez.org/cases/2018/18-422 (Accessed: 6 July 2022).
- Simonyan, K. and Zisserman, A. (2015) 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. arXiv. Available at: http://arxiv.org/abs/1409.1556 (Accessed: 30 June 2022).
- *sklearn.model_selection.GridSearchCV* (no date) *scikit-learn*. Available at: https://scikit-learn/stable/modules/generated/sklearn.model_selection.GridSearchCV.html (Accessed: 1 July 2022).
- *sklearn.model_selection.RandomizedSearchCV* (no date) *scikit-learn*. Available at: https://scikit-learn/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html (Accessed: 1 July 2022).
- Smith, R. (2019) 'Duke Mathematics Has Its Day in Court', 25 March. Available at: https://today.duke.edu/2019/03/duke-mathematics-has-its-day-court (Accessed: 6 July 2022).
- Stephanopoulos, N.O. and McGhee, E.M. (no date) 'Partisan Gerrymandering and the Efficiency Gap', *The University of Chicago Law Review*, p. 70.
- Team, K. (no date) *Keras documentation: Keras Applications*. Available at: https://keras.io/api/applications/ (Accessed: 7 July 2022).
- *TIGER/Line Shapefiles* (no date) *Census.gov*. Available at: https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.html (Accessed: 6 July 2022).
- Ye, A. (2020) *Turning Non-Image Data into Images for Classification is Surprisingly Effective*, *Medium*. Available at: https://towardsdatascience.com/turning-non-image-data-into-images-for-classification-is-surprisingly-effective-70ce82cfee27 (Accessed: 29 June 2022).

Q&A

Backup

Evaluation – Multiclass Results

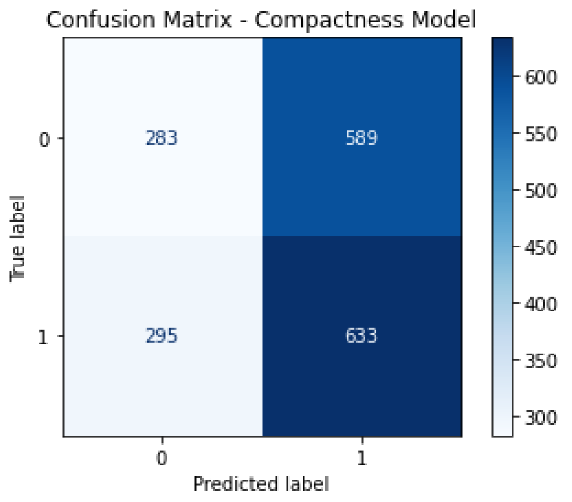
- Baseline accuracy: 50%
- Best multiclass model test accuracy: 70.8%
  - Low, Medium, and High classes labeled by EGS metric
  - Low: $[-0.5, 0.5]$
  - Medium: $[\mu - 1\ std, -0.5] \cup [0.5, \mu + 1\ std]$
  - High: $[-1, \mu - 1\ std] \cup [\mu + 1\ std, 1]$
  - Other labeling methods caused massive drops in accuracy
- More sensitive to size of data



	precision	recall	f1-score	support
high	0.69	0.77	0.73	602
low	0.00	0.00	0.00	297
medium	0.72	0.90	0.80	901
accuracy			0.71	1800
macro avg	0.47	0.56	0.51	1800
weighted avg	0.59	0.71	0.64	1800

Evaluation – Compactness Results

- Baseline accuracy: 51.5%
- Statewide model test accuracy: 50.9%
- Very poor accuracy
- Illustrates difficulty of using compactness metrics for labeling data

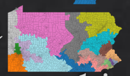


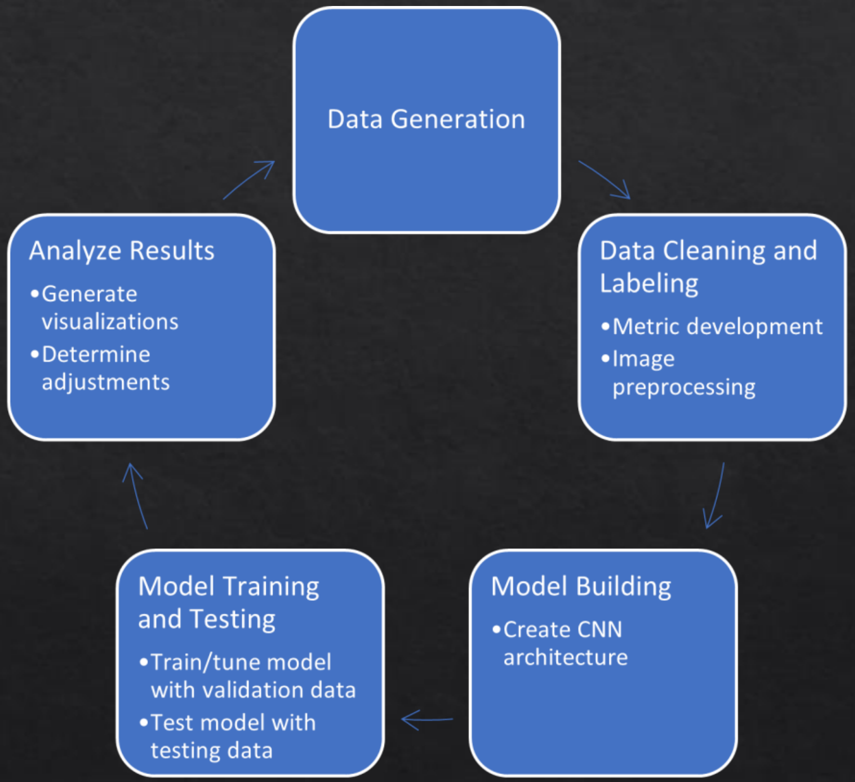
	precision	recall	f1-score	support
False	0.49	0.32	0.39	872
True	0.52	0.68	0.59	928
accuracy			0.51	1800
macro avg	0.50	0.50	0.49	1800
weighted avg	0.50	0.51	0.49	1800

Evaluation – Future Work

- Explore metrics and labeling process even more
- Replace and/or build own data generation module
- Add more demographic/numerical data
- Incorporate voter registration into data and/or metric
- Generate data and build models for different state(s)
  - Synthetic "grid state" as well
- Spend more time with the statewide models
- Use recall as model metric instead of accuracy
- Evaluate feature importance scores within the models
- Explore possibilities for unsupervised learning

Development and Implementation

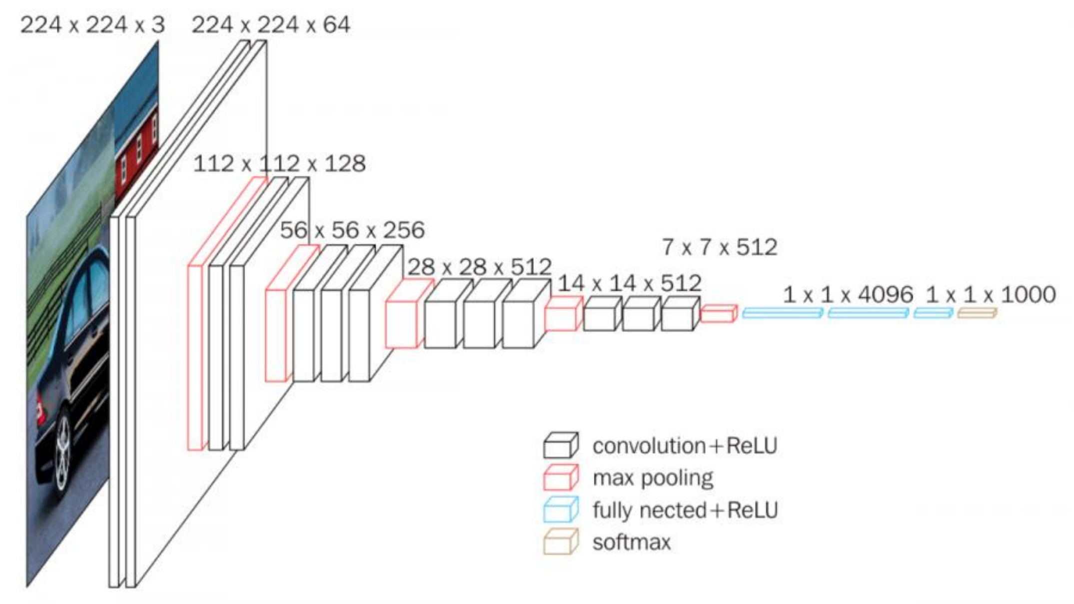
- Standard predictive modeling development cycle
- Trial-and-error in beginning stages
- Transfer learning with VGG16
- Initially, two models using:
  - District image data 
  - State image data 
  - Numerical demographic data in both
    - Total district population
    - White population
    - Black population
    - Area of district
- Each will use different sets of metrics to label data
- Will adjust future models based on analysis of initial results



Development and Implementation - CNN

- Transfer learning with VGG16 CNN initially
- Adjust model as needed based on initial model accuracies
- Additional pre-built CNNs to consider
  - VGG19
  - ResNet
  - EfficientNet
  - Etc… (Keras CNNs)
- Must develop strategy to combine the numerical demographic data into CNN

VGG16 Architecture



(ul Hassan, 2018)

Outputs - “Basic” Model Architecture