Kate Radford

11537868

CptS 315 HW 4

|Q1| (10 points) Suppose x = (x1, x2, · · · , xd) and z = (z1, z2, · · · , zd) be any two points in a high-dimensional space (i.e., d is very large). We know that the computation of nearest neighbors is very expensive in the high-dimensional space. Discuss how we can make use of the above property to make the nearest neighbors computation efficient?

The left hand side of the inequality is the squared distance between the average of feature values of x and z. By finding the average of feature values, this essentially reduces each point down to one dimension. This increases efficiency because instead of calculating the complex Euclidean distance between each set of points, you can compute for each point then plug into the simpler equation, to get ‘distance’. Using the left side equation instead of Euclidean distance, we can still sort and determine knn as we would normally would after distance calcuations.

|Q2| (10 points) We know that we can convert any decision tree into a set of if-then rules, where there is one rule per leaf node. Suppose you are given a set of rules R = {r1, r2, , rk}, where ri corresponds to the i th rule. Is it possible to convert the rule set R into an equivalent decision tree? Explain your construction or give a counterexample.

Yes, it is possible. Each rule in R will give us attribute conditionals and the classification. Assuming the rule set R is not in order (of tree construction), we need to decide that for ourselves. To do this, I would grab all the attributes and possible values from the rule set. Then, I would construct the first node from the attribute with the most values/branches. After that, for each branch, I would grab all rules containing that attribute, value pair and pick the most reoccurring attribute as the next node. This would continue down each branch until a rule defines the class for that leaf.

|Q3| (10 points) Suppose you are given 7 data points as follows: A = (1, 1); B = (1.5, 2.0); C = (3.0, 4.0); D = (5.0, 7.0); E = (3.5, 5.0); F = (4.5, 5.0); and G = (3.5, 4.5). Manually perform 2 iterations of K-Means clustering algorithm (slide 22 on clustering) on this data. You need to show all the steps. Use Euclidean distance (L2 distance) as the distance/similarity metric. Assume number of clusters k=2 and the initial two cluster centers C1 and C2 are B and C respectively.

Distance Computations:

A-C1=sqrt[(1.5-1)^2+(2-1)^2] = 1.1

A-C2=sqrt[(3-1)^2+(4-1)^2] = 3.6

B-C1=sqrt[(1.5-1.5)^2+(2-2)^2] = 0

B-C2=sqrt[(3-1.5)^2+(4-2)^2] = 2.5

C-C1=sqrt[(1.5-3)^2+(2-4)^2] = 2.5

C-C2=sqrt[(3-3)^2+(4-4)^2] = 0

D-C1=sqrt[(1.5-5)^2+(2-7)^2] = 6.1

D-C2=sqrt[(3-5)^2+(4-7)^2] = 3.6

E-C1=sqrt[(1.5-3.5)^2+(2-5)^2] = 3.6

E-C2=sqrt[(3-3.5)^2+(4-5)^2] = 1.1

F-C1=sqrt[(1.5-4.5)^2+(2-5)^2] = 4.2

F-C2=sqrt[(3-4.5)^2+(4-5)^2] = 1.8

G-C1=sqrt[(1.5-3.5)^2+(2-4.5)^2] = 3.2

G-C2=sqrt[(3-3.5)^2+(4-4.5)^2] = 0.7

|  |  |  |  |
| --- | --- | --- | --- |
| Point | Distance to C1 | Distance to C2 | Cluster |
| A (1,1) | 1.1 | 3.6 | C1 |
| B (1.5,2) | 0 | 2.5 | C1 |
| C (3,4) | 2.5 | 0 | C2 |
| D (5,7) | 6.1 | 3.6 | C2 |
| E (3.5,5) | 3.6 | 1.1 | C2 |
| F (4.5,5) | 4.2 | 1.8 | C2 |
| G (3.5,4.5) | 3.2 | 0.7 | C2 |

C1 Centroid = (1+1.5/2, 1+2/2) = (1.25, 1.5)

C2 Centroid = (3+5+3.5+4.5+3.5/5, 4+7+5+5+4.5/5) = (3.9, 5.1)

A-C1=sqrt[(1.25-1)^2+(1.5-1)^2] = 0.5

A-C2=sqrt[(3.9-1)^2+(5.1-1)^2] = 5

B-C1=sqrt[(1.25-1.5)^2+(1.5-2)^2] = 0.5

B-C2=sqrt[(3.9-1.5)^2+(5.1-2)^2] = 3.9

C-C1=sqrt[(1.25-3)^2+(1.5-4)^2] = 3

C-C2=sqrt[(3.9-3)^2+(5.1-4)^2] = 1.4

D-C1=sqrt[(1.25-5)^2+(1.5-7)^2] = 6.6

D-C2=sqrt[(3.9-5)^2+(5.1-7)^2] = 2.1

E-C1=sqrt[(1.25-3.5)^2+(1.5-5)^2] = 4.1

E-C2=sqrt[(3.9-3.5)^2+(5.1-5)^2] = 0.4

F-C1=sqrt[(1.25-4.5)^2+(1.5-5)^2] = 4.7

F-C2=sqrt[(3.9-4.5)^2+(5.1-5)^2] = 0.6

G-C1=sqrt[(1.25-3.5)^2+(1.5-4.5)^2] = 3.7

G-C2=sqrt[(3.9-3.5)^2+(5.1-4.5)^2] = 0.7

2nd Iteration:

|  |  |  |  |
| --- | --- | --- | --- |
| Point | Distance to C1 | Distance to C2 | Cluster |
| A (1,1) | 0.5 | 5 | C1 |
| B (1.5,2) | 0.5 | 3.9 | C1 |
| C (3,4) | 3 | 1.4 | C2 |
| D (5,7) | 6.6 | 2.1 | C2 |
| E (3.5,5) | 4.1 | 0.4 | C2 |
| F (4.5,5) | 4.7 | 0.6 | C2 |
| G (3.5,4.5) | 3.7 | 0.7 | C2 |

C1 Centroid = (1+1.5/2, 1+2/2) = (1.25, 1.5)

C2 Centroid = (3+5+3.5+4.5+3.5/5, 4+7+5+5+4.5/5) = (3.9, 5.1)

|Q4| (25 points) Please read the following two papers and write a brief summary of the main points in at most FOUR pages. Ten simple rules for responsible big data research. PLoS Computational Biology 13(3) (2017)

Ethical questions have been raised in big data research due to the increasing size and complexity of datasets able to be collected. Research agendas are focusing more on sensitive parts of human behavior and the tools for big data mining are more and more interworked into our daily lives. The scope and pure amount of big data collected has now left human participants susceptible to harm. To guide responsible research, this paper presents ten rules addressing the ethical issues that arise in big data research. These rules will envelope the goal of minimizing potential harm on the human subjects which big data is collecting research.

Rule one, acknowledge that data are people and can do harm. It is important to recognize that most data represents and/or impacts people. The fear here is that innocuous datasets are used to categorize, map, change rules that inadvertently and negatively hurt minority people or groups. In even easier words, using neutral data in research can still produce discriminatory outcomes.

Rule two, recognize that privacy is more than a binary value. Privacy is contextual and situational. People often perceive something to be creepy or a breach of privacy if they don’t expect that data to be utilized. Some data is okay to use for one purpose, but a breach of privacy if used for another purpose. For example, land maps used to assert land rights vs used to identify exploitation opportunities. It is important to make sure the data creators’ consent to the use of their data.

Rule three, guard against the reidentification of your data. You want to make sure your data personas identities cannot be revealed with the combination of another variable or dataset. The anonymization of the data is difficult as vulnerable points are hidden as irrelevant or harmless datapoints such as battery usage. Vulnerabilities for unexpected reidentification also lie in new features created in the future. For example, locations on social media could provide a way to determine home addresses.

Rule four, practice ethical data sharing. Consent scope is determined in the contracts, and while it’s easy to ask for a broad consent for easy sharing, one should consider things like privacy breaches, reidentification issues, etc. for the human data creators when drafting the study. Unfortunately, this practice is not typical today in the gathering of data. Often, it is impossible to get informed consent because researchers don’t get the data until after it’s collected by state agencies, private businesses and telecom firms.

Rule five, consider the strengths and limitations of your data, big does not automatically mean better. Context is important to include in dataset research, used to determine when the data and analysis are working or not. Data can represent/hold multiple meanings which also impacts the interpretation of findings.

Rule six, debate the tough, ethical choices. While institutional review boards govern research involving human participants, researchers often still come across situations not addressed by the IRB and should discuss issues with peer groups before proceeding. Scrutinizing ethical issues is an essential part of professional development in order to establish a community of responsible researchers.

Rule seven, develop a code of conduct for your organization, community or industry. Instead of allowing ethics to be an optional afterthought, develop codes of conduct for your group. This will make researchers more successful as the rules are utilized in daily practice instead of only considered in damage control.

Rule eight, design your data and systems for auditability. Auditability means clear documentation of decisions or dataset information discarded as not presently relevant. These documentations become useful when double checking work or critical re-assessment. Developing a automated system for testing processes is recommended as it will only strengthen research.

Rule nine, engage with the broader consequences of data and analysis practices. Researchers have so much power in the interest they select to collect and analyze. Instead of focusing on research areas for traditional success and money, imagine the impact researchers could have if they focused their efforts on big issues like climate change, renewable sources, etc.

Rule ten, know when to break these rules. A couple obvious circumstances in which to break these rules are for example, a natural disaster or public health emergency. The key component is to make sure the emergency is not a convenient justification, but the collection of data without informed consent is essential to save lives in a time-sensitive crisis.

Interventions over Predictions: Reframing the Ethical Debate for Actuarial Risk Assessment. Proceedings of Machine Learning Research (PMLR), 81:62-76, 2018

Critics have begun to claim that the techniques in actuarial risk assessments may reproduce existing patterns of discrimination and historical biases already existing in the data. There are debates over the fairness and accuracy of the use of typically discriminated characteristic proxies. Instead of using machine learning for predicting risks, why not use it for risk mitigation? In 2016, a risk assessment tool used by the US criminal justice system was claimed as bias by a team of investigative journalists. This created a debate that dove into the increasing use of algorithmic decision-making aids in the courts. Artificial intelligence, while new and ‘cutting edge’, is often iterations on old actuarial decision-making practices. This means that bias can still work its way into AI.

There have been calls for transparency of calculations by decision making tools in the courts, as defendants only get to see their end scores and they have a right to due process. Some human error still impacts the end score as judges and practitioners often misapply/interpret scores. There are also some purposeful strategies used to manipulate end scores, like inappropriately referencing a score for a decision it was not intended to inform. Researchers have argued that risk assessments are not very accurate to start, largely failing in accuracy disproportionately across racial lines.

The first generation of risk assessment tools were criticized for being too subjective. The second generation focused on static factors like age, criminal history. This lead to a shift in the values the risk assessments supported, placing heavier weight on prior criminal history. Because of this, there was also a shift in the way the criminal justice system responded to offender risk. The third generation included “criminogenic needs” factors like employment status and history of drug abuse. Currently, risk assessments carry two purposes: prediction and reduction oriented assessments. Prediction oriented assessments are of interest during the pretrial stage.

|Q5| (25 points) Please go through the excellent talk given by Kate Crawford at NIPS-2017 Conference on the topic of “Bias in Data Analysis” and write a brief summary of the main points in at most FOUR pages.

We are at an inflection point where ML is rapidly expanding into our daily lives. There are problems arising like bias in the AI, as seen in the news. Some racially related terms are viewed as negative, while ‘white’ terms are seen as positive. The history of our racial bias is deep rooted even into our machine learning systems. This ‘core problem’ needs to be fixed as those affected are increasing daily. Structural bias is a ‘socio-technical’ issue and not nearly solved. Bias can lead to distrust in artificial intelligence in the future. There is some confusion on what bias really means. Bias is judgement based on preconceived notions or prejudices. And machine learning systems can me unbiased in code but produce legally biased results.

Allocative harm is when a system allocates or withholds certain groups from an opportunity or resource. For example, mortgage applications being denied for women or certain racial groups. While ML systems represent our society, they don’t deal with the allocation, but they does partake in representation.

Representation bias is at root, cultural. Systems that represent our society but don’t deal out resources are representational harms. Harms of representation include stereotypes, recognition (facial recognition bias between races), denigration (photo mislabeling due to race), under-representation (google images for ‘CEO’ are all white males) and ex-nomination.

Classification is always a product of its time, a reflection of culture. This is important because misclassification will negatively impact those who are mis-classified. Classification models trained on non-strategic (not compiled with purpose) training photos will favor while males simply because the majority of photos on the web are white males. Classification systems show signs of political and social struggle. For example, china developed a facial recognition that claims to sort criminals vs non criminals solely based on facial points. Solutions include fairness forensics to self-evaluate our systems, and utilization of interdisciplinary groups while creating ML systems. All of this builds to the question, are there some things we just shouldn’t build?

|Q6| (20 points) Please read the following two papers and write a brief summary of the main points in at most THREE pages. Hidden Technical Debt in Machine Learning Systems. NIPS 2015: 2503-2511

An ‘uncomfortable’ trend has emerged in machine learning where developing and deploying ML systems are difficult and expensive to maintain. This is called technical debt, where the goal isn’t to add new functionality, but allow future improvements. Technical debt is a metaphor about how moving fast in software creates long term cost as a downside. They say that ML systems are especially good at gathering technical debt, with lots of maintenance problems and ML specific issues on top.

Encapsulation and modular design are best to keep code maintainable, but enforcing strict abstraction boundaries for ML systems is difficult. There are a bunch of ways a loss of boundaries will add technical debt in machine learning systems, including entanglement, correction cascades, and undeclared consumers. Entanglement refers to the mixing of signals. This is bad because it makes isolation of improvements impossible. Correction cascades means with one fix, it creates more down the line until a possible improvement deadlock. Undeclared consumers are silently using the output as an input to another system without access controls. This issue is also called visibility debt.

A key contributor to technical debt is dependency debt. It is hard to detect and very easy to create large data dependency chains on accident. Some input signals are unstable, where their behavior changes over time. Updates to the input signal can be made at any time and is dangerous because it could have arbitrary detrimental effects in the system that are costly. One strategy for unstable data is to make a versioned copy of the signal to use until the next version has been vetted for use. Underutilized dependencies are packages that are unneeded, for example, legacy features, bundled features, and correlated features. Leaving old schemes in the system creates more problems later, like if there are still dependencies hidden.

ML systems often influence their own behavior as they update. This is called analysis debt. Also called feedback loops, there are direct and hidden loops. Direct loops is when the model directly influences its training data. A strategy used here is randomization or isolation of data from the model. Hidden loops are when two systems influence each other indirectly. That can be hard to track down as they can be completely disjoint systems.

The world is always changing, so the ML systems have to change to keep up with us. This creates an ongoing maintenance cost. Unit testing is preferable, but not possible in such an everchanging environment. Instead, a live monitoring of system behavior is needed to track that the system is working appropriately. Within the live monitoring, they pay close attention to prediction bias, action limits and up-stream producers. These invariants are monitored closely and set up with automated response if needed.

The ML test score: A rubric for ML production readiness and technical debt reduction. BigData 2017: 1123-1132

The issue of ML reliability has come up as machine learning systems are assigned increasingly larger roles in our world. Important strategies to improve reliability include testing and monitoring. ML systems are so complex that testing is becoming difficult. The paper will discuss a set of tests and a scoring system to determine if a system is ready for production.

For testing features and data, one cannot rely on unit testing and integration tests as you normally would. Instead, they suggest the following tests. First, construct a schema of feature expectations to avoid an anchoring bias. Second, understand the value of each feature independent of other features. Third, measure costs of each feature to make sure it measures up with the predictive benefit. Fourth, enforce meta-level requirements programmatically. Fifth, ensure the data pipeline has appropriate privacy controls during new feature development. Sixth, increase speed in moving a new feature idea into production, as it only improves the system. And lastly, verify input feature code is tested because they are near impossible to detect once they start the data generation.

Tests for model development. Don’t skip code review, make sure every model spec gets a code review and repository for proper version control. Understand the relationship between offline proxy metrics and the real impact metrics in order to verify they correlate and will result in a better system. Use a grid search or other hyperparameter search strategy to tune and uncover hidden reliability issues. Data is ever changing, also called non-stationary. If the ML system can no longer sufficiently train and deploy, then that model is stale and needs to be updated to account for external world changes. To determine model quality, slice the data set. Make sure your model is sufficient for all types of data slices. Overlooked biases in training data will influence the whole system behavior, combat this by running tests to examine input features and their correlation with protected user categories.

Tests for infrastructure. For the first test, make sure training is reproducible. Then, unit test the model specification code for bugs. Third, integration test the full ML pipeline. The ML pipeline consists of gathering training data, feature generation, model training, model verification, then deployment. You don’t want any errors that could affect later stages in the pipeline. Fourth, make sure model quality is validated before moving onto production. Fifth, debug odd model behavior by observing step by step computation on a single example. Sixth, utilize the canary process to test models before entering the production stage, to make sure the newer model code is compatible with the older serving system. Lastly, save all code changes to ensure models can be reverted to a previous version in emergency situations.