Characterizing Optimal Uncertainty Sets for Linear Regression

Charlie W. Chimento
Technology and Policy Program
Massachusetts Institute of Technology
cchiment@mit.edu

Desiree Waugh
MBAn
Massachusetts Institute of Technology
dwaugh@mit.edu

Abstract

Robust optimization adheres to a set-based, deterministic approach to account for uncertainty, whereas stochastic optimization assumes some underlying probability distribution. A critical component of robust optimization, then, is designing uncertainty sets that formalize a priori knowledge. In this work, we seek to understand what characteristics of a dataset render it amenable to specific uncertainty sets for linear regression. This included building features for 156 datasets, and using an Optimal Classification Tree (OCT) to predict the optimal uncertainty set. Randomness of the data set split caused significant variation in the features that the OCT split on, so multinomial logistic regression was implemented using features frequently selected by the OCT.

1 Problem Setup

At the heart of machine learning is generalizing to unseen data. Robust optimization attempts this by modeling random variables as uncertain parameters belonging to a convex uncertainty set. This approach achieves "robustness" by immunizing against adversarial perturbations in the data. In the case of linear regression, this problem is recognized in the general form

$$\min_{\beta} \max_{\Delta \in U} g(\mathbf{y} - (\mathbf{X} + \mathbf{\Delta})\beta), \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{X} \in \mathbb{R}^{nxp}$ with g being some choice of a norm. The uncertainty set, $U \subseteq \mathbb{R}^{nxp}$, formalizes the practitioner's assumptions about the uncertainty in the data matrix. The inner maximization problem reflects the emphasis on training against the worst possible errors, as the permutation matrix Δ is chosen to maximize what the outer problem seeks to minimize–hence adversarial.

1.1 Brief History

Early attempts to account for uncertainty set parameters were to find those equal to the worst-case value. While this approach achieves robustness, it comes at a cost of being overly conservative. Ben-Tal, Nemirovski, and El-Ghaoui addressed this by bounding parameters to ellipsoidal uncertainty sets [1]. This is unfavorable from a computational perspective, however, because a robust framework with ellipsoidal uncertainty sets transforms linear programs to second-order cone problems [5]. Other work improved this by developing a robust optimization approach with polyhedral uncertainty sets, thus retaining the linearity of the constraints [5]. Successful approaches in maintaining the linearity of programs in their robust formulation leverage strong duality. Another foray of robust optimization that has gained attention is consideration of dynamic decision-making. In this setting, parameters are realized over time, and decisions can be made in a multi-stage fashion. This project, however, focuses on static robust optimization.

MIT 9.520/6.860: Statistical Learning Theory and Applications (Fall 2016).

1.2 Uncertainty Sets

The space of possible uncertainty sets is bounded by some matrix norm, and in the case of singular values, a standard norm. Norms that are often used are Frobenius, Induced, and p-Spectral.

$$U_{F(q)} = \{\Delta \in \mathbb{R}^{nxp} : ||\Delta||_{F-q} \leq \rho\} \tag{2}$$
$$U_{(r,q)} = \{\Delta \in \mathbb{R}^{nxp} : ||\Delta||_{r,q} \leq \rho\} \tag{3}$$
$$U_{\sigma_p} = \{\Delta \in \mathbb{R}^{nxp} : ||\mu(\Delta)||_p \leq \rho\} \tag{4}$$

For the Frobenius norm, $||\Delta||_{F-q} := \left(\sum_{i,j} |\Delta_{ij}|^q\right)^{1/q}$, for the induced norm $||\Delta||_{r,q} := \max_x \frac{||\Delta x||_q}{||x||_r}$, and the Schatten norm is a vector p-norm on the eigenvalues of the matrix, $\mu(\Delta)$ [2].

1.3 Choosing an Uncertainty Set

Variations of these uncertainty sets render flexibility to implicitly design a model to be robust against particular kinds of perturbations. Whereas the Frobenius matrix norm implies unstructured uncertainty throughout the data matrix, the uncertainty set constrained by $U_{(1,2)}$, is a reformulation of Lasso [1]. Given a data set, the question of interest becomes, *how do I choose my uncertainty set*? In this work we judge the performance with Least Absolute Deviations on the downstream task of linear regression.

2 Importance

At the heart of machine learning is a balance of enforcing a priori intuition against randomness. Robust optimization achieves this with uncertainty sets. Thus, it is important for decision makers to understand what the uncertainty set implicitly formalizes and its connection to other regularization techniques. A useful example of this is found in Artzner et al, which provided a set of axioms that gives explicit control to incorporate risk preferences by formalizing a class of coherent risk measures [8].

In this field, a plethora of efforts have worked to propose approaches tailored to specific settings. As briefly surveyed above, these settings include single-stage and multi-stage decisions, and settings with uncertainty in parameter estimation with the parameter as either static or stochastic. The variety of approaches provides flexibility to the decision maker to pick the robust optimization formulation that properly locates uncertainty and is tractable. These variations trace an insistence to leverage historical observations of random variables as direct inputs.

This work focuses on the static, linear regression problem with uncertainty in the data. It seeks to provide an empirical foothold to extend theory by suggesting how to interpret characteristics of a dataset in order to predict the optimal uncertainty set for linear regression.

3 Data

The meta-dataset was created from built-in R datasets using the RDatasets package in Julia. Datasets with categorical variables were excluded to avoid high-dimensionality inherent with using binary dummy variables for categorical entries. Also excluded were datasets with fewer than two features (to enable calculation of correlation between different features), and datasets with fewer than 40 observations. The R datasets spanned a variety of domains including economics, health, and biological systems.

For the purposes of optimizing over a particular uncertainty set to find β, the last column in the R dataset was treated as the independent variable. For this research problem, it did not matter which specific column was treated as the independent variable, because the goal was to find the optimal uncertainty set for any given linear regression problem. Missing data was imputed using Optimal k-NN imputation [4], which uses a formulation blind to downstream tasks.

4 Methods

If the goal is to pick the uncertainty set to minimize absolute deviations of linear regression given a data set, a dataset must be constructed with appropriate labels. The process involved four steps:

1. Develop code to find the optimal uncertainty set (of a specific type, such as $U_{(1,2)}$) for a given dataset
2. Develop code to find which specific type of uncertainty set (i.e. $U_{(1,2)}$, $U_{(2,\infty)}$, etc.) minimizes absolute deviations in linear regression
3. Generate features to be calculated for each dataset
4. Evaluate the performance of an OCT in predicting the optimal type of uncertainty set, given the generated features

4.1 Finding the optimal uncertainty set

The metric to be minimized was Mean Absolute Error, so as not to bias towards what has been shown to be equivalent regularization formulations. For example, in [2], Bertismas and Copenhaver showed that

$$\min_{\beta} \max_{\Delta \in U_{(q,p)}} ||\mathbf{y} - (\mathbf{X} + \Delta)\beta||_p = \min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||_p + \lambda ||\beta||_q \tag{5}$$

$$\min_{\beta} \max_{\Delta \in U_{\sigma_p}} ||\mathbf{y} - (\mathbf{X} + \Delta)\beta||_2 = \min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||_2 + \lambda ||\beta||_2 \tag{6}$$

$$\min_{\beta} \max_{\Delta \in U_{F(p)}} ||\mathbf{y} - (\mathbf{X} + \Delta)\beta||_p = \min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||_p + \lambda ||\beta||_{p*} \tag{7}$$

where $p \in [1, \infty)$ satisfies $\frac{1}{p} + \frac{1}{p*} = 1$ for $p* \in [1, \infty]$ for norm duality with the Frobenius uncertainty matrix. These equivalent formulations were leveraged because the right-hand side is readily solvable via JuMP.

Seven uncertainty sets were selected for the classification problem.

- $||\Delta||_{F-1}$
- $||\Delta||_{F-2} = ||\Delta||_{\sigma_2}$
- $||\Delta||_{F-\infty}$
- $||\Delta||_{(1,2)}$
- $||\Delta||_{(2,1)}$
- $||\Delta||_{(2,\infty)}$
- $||\Delta||_{(\infty,2)}$

The right hand side of the expressions in (5,6,7) are amenable to optimization via JuMP. The λ hyperparameter was chosen per performance on a validation set, with the same range of λ values used across all uncertainty sets.

In this fashion, the labels for an optimal uncertainty set were assigned according to the uncertainty set that achieved the lowest MAE.

4.2 Feature Engineering

Once a method was established to identify the optimal uncertainty set, features were generated for each dataset. Chosen features were

- R^2 for $\hat{\beta}$ coming from Ridge Regression
- Percentage of missing data points in $\mathbf{y}$
- Percentage of missing data points in $\mathbf{X}$
- Number of observations

- Number of features
- Percentage of outliers in $\mathbf{y}$
- Mean Pairwise Correlation (transformed mean of Fischer's Z-statistic)
- Mean normalized variance of $\mathbf{X}$
- Median normalized variance of $\mathbf{X}$
- Range of normalized variance in $\mathbf{X}$
- Percentage of features that are correlated above a certain threshold in $\mathbf{X}$
- Variance in $\mathbf{y}$ divided by mean in $\mathbf{y}$

4.3 Feature discussion and justification

If a primary benefit of OCTs over Random Forests and Boosted Trees are their interpretability, then it is important to understand what each generated feature represents.

The intuition of using R^2 was to proxy how amenable a dataset is to a linear model. But calculating R^2 involves $\hat{\mathbf{y}}$, and that necessitates a $\hat{\beta}$. Statistical learning theory shows us that different approaches to estimating β carry different biases. For instance, using OLS biases a solution towards principal components. The normal equations, $\beta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$, yield the $\hat{\beta}$ that minimizes empirical squared loss, but if $d >> n$, taking the inverse of $\mathbf{X}^T\mathbf{X}$ becomes less computationally feasible. Because the datasets ranged from $n >> d$ to $d << n$, it was also not feasible to use $\beta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ (1) if $n > d$, and $\beta = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}$ (2) if $n < d$. While (1) minimizes squared loss, (2) minimizes the squared sum of the components of s.t. $Y = X$. So the R^2 feature would vary depending on whether $n > d$ or $n < d$.

For these reasons, ridge regression was chosen because it would scale well across variable sizes of n and d using the Gurobi solver through JuMP. The ρ was selected (from a consistent range) that minimized MSE on a hold-out validation set.

Intuitively, the effect of missing data points on MAE would be more drastic if observations with missing features were discarded rather than imputed via kNN, but the data was imputed so as to conserve the number of available datasets with continuous-valued features.

4.4 Optimal Classification Tree

Interpretability is key in this effort, because the goal is to understand what characteristics of a dataset might lend themselves to predicting the optimal uncertainty set. Optimal Classification Trees (OCTs) provide interpretable diagrams of which variables the tree splits on. Like Decision Trees, Random Forests, and Boosted Trees, OCTs are nonlinear classifiers. Decision Trees have the downside of being greedy, which algorithms help account for with post-pruning. Random Forests leverage the principle of aggregated wisdom but are similarly unintepretable because they make decisions per averaging the vote of an ensemble of trees. Thus Optimal Classification Trees were the prime candidate for predicting the optimal uncertainty set for linear regression given a dataset.

4.5 Summary of Method

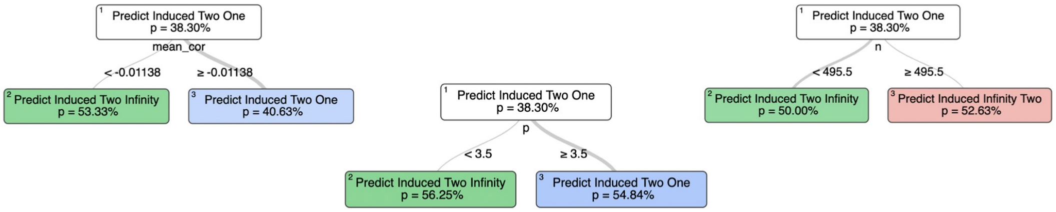
- Find the uncertainty set that minimizes the mean absolute error for linear regression on 156 datasets pulled from the R-repository
- Generate features for each dataset (mean pairwise correlation, etc...)
- Train an Optimal Classification Tree and evaluate its performance in predicting the best uncertainty set

5 Results

Initial results showed performance on-par or worse than a baseline predictor that chooses the most common label in the training set. In order to improve performance, the number of possible classifications was reduced to the three most commonly selected ($U_{(2,1)}$, $U_{(2,\infty)}$, $U_{(\infty,2)}$). The variables that the OCT split on changed due to randomness in the data split, so a multinomial logistic regression model was built to predict the three most common classes using three variables highlighted by the OCT. A geometric comparison of the feasible set for β as inferred by the uncertainty set equivalent formulations (5,6,7) elucidates the nature of the solution for the uncertain parameter, β , a given uncertainty set biases towards.

5.1 Optimal Classification Tree Behavior

Optimal Classification Trees, although they have the potential for hyperplane splits, are more interpretable when each split is performed on a single variable. Choosing *max depth* and *min bucket size* with cross-validation, the trained tree tended to return with a selected depth of 2. At a smaller depth, the selection of variables that the OCT decides to split on is telling. However, the selection of variables that the OCT split on were very sensitive to the data split. The figure below shows examples of three different depth-2 trees.



1 Predict Induced Two One
p = 38.30%
mean_cor
< -0.01138
≥ -0.01138
2 Predict Induced Two Infinity
p = 53.33%
3 Predict Induced Two One
p = 40.63%
1 Predict Induced Two One
p = 38.30%
p
< 3.5
≥ 3.5
2 Predict Induced Two Infinity
p = 56.25%
3 Predict Induced Two One
p = 54.84%
1 Predict Induced Two One
p = 38.30%
n
< 495.5
≥ 495.5
2 Predict Induced Two Infinity
p = 50.00%
3 Predict Induced Infinity Two
p = 52.63%

Figure 1: Optimal Classification Trees splitting on different dataset features to predict the optimal uncertainty set

5.2 OCT Performance

Using all 12 of the generated features, the out-of-sample performance was slightly worse than the baseline in the case of choosing among 3 uncertainty sets. When choosing among all 7 constructed uncertainty sets, the performance of the OCT was also worse than the baseline model. Table 1 reports the performance.

Table 1: OCT Prediction Performance as Measured by Mean Accuracy and Accuracy Variance Between Trials

trials = 100	features = all				
# Labels	In-Sample Acc.	Test Acc.	In-Sample Acc. Var.	Test Acc. Var.	Baseline Acc.
3	58.10%	40.79%	0.46%	0.61%	42.30%
7	41.51%	23.83%	0.59%	0.36%	27.20%

5.3 Further Results with Logistic Regression

While the OCT did not yield strong performance, it did consistently highlight a few variables as important in determining the optimal uncertainty set. Thus it seemed a natural extension to explore the accuracy of a multinomial logistic regression model regressing on these highlighted features. The results beat the baseline out-of-sample accuracy. For comparison, a logistic regression model was trained and tested using all the generated features and the mean performance was 32.80% accuracy, suggesting that the OCT served a purpose in highlighting the generated features which are indicative of which uncertainty set would be optimal for minimizing MAE on linear regression. Table 2 summarizes the performance of the logistic regression using the features identified by the OCT.

Table 2: Logistic Regression Prediction Performance as Measured by Mean Accuracy and Accuracy Variance Between Trials

trials = 100 features = n, p, mean pairwise corr.					
# Labels	In-Sample Acc.	Test Acc.	In-Sample Acc. Var.	Test Acc. Var.	Baseline Acc.
3	49.83%	42.97%	0.10%	0.41%	42.30%

5.4 Geometric Interpretation

The uncertainty sets that most often returned the best out-of-sample performance were $U_{(1,2)}$, $U_{(2,\infty)}$, and $U_{(\infty,2)}$. This translates to an l_1, l_2 and l_∞ norm on β, respectively. A geometric perspective lends insight on the type of solutions for β that each of these uncertainty sets biases towards. The figure below visualizes the shape of the feasible region for two-dimensions of β. Note that the optimal solution will lie at the corners for l_1 and l_2, or in the case of the circle, anywhere along the circle border.

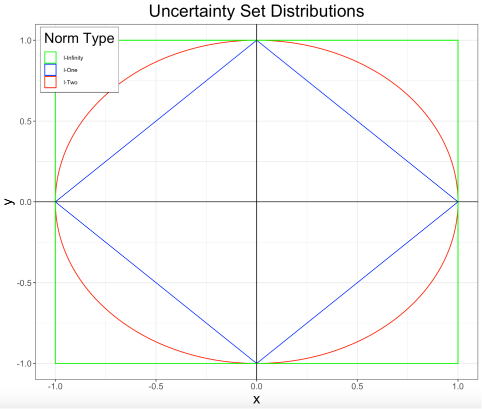


Figure 2

The geometry demonstrates the bias of different norms on β towards certain solutions, each with a distinct nature. $U_{(2,\infty)}$ will almost certainly not be sparse in β and $U_{(\infty,2)}$ uncertainty sets will tend to produce β without bias towards either sparsity or density.

A natural question, then, is *what characteristics of a dataset would suggest that sparsity as opposed to just minimizing each component of β would be preferable*? If the simulation were to be run again, it would be interesting to generate features characterizing a dataset that are known to be more amenable to sparse as opposed to dense βs and see whether the OCT uses this feature in classifying the equivalent optimal uncertainty set.

6 Conclusions

Identifying the optimal uncertainty set is an inherently challenging problem because there are many degrees of freedom, from selecting the performance parameter, to dealing with missing values, to randomness in the split for training, validation, and testing.

Further areas of exploration include uncertainty tests built on hypothesis tests. Previous research has shown a way to build uncertainty sets for continuous, independent features using the Kolmogorov-Smirnov test with probabilistic guarantees [3].

Additionally, it was difficult to build a strong machine learning model using only 156 observations. It would be instructive to repeat the procedure with a much larger dataset compiled from a variety of sources and domains.

7 Contributions

Charlie Chimento

- Wrote final paper
- Research on theory for final paper
- Code for feature engineering

Desiree Waugh

- Code to optimize over uncertainty sets and find the optimal uncertainty set for a given dataset
- Code to read in R Datasets and create meta-dataset with 156 observations
- Code for OCT and multinomial logistic regression

References

[1] Ben-Tal, A. & Goryashko, A. & Guslitzer, E. & Nemirovski, A. (2003) Adjustable Robust Solutions of Uncertain Linear Programs, *Springer-Verlag*, Math. Proram. Ser. A 99:351-376.

[2] Bertsimas, D. & Copenhaver, M. (2018) Characterization of the Equivalence of Robustification and Regularization in Linear and Matrix Regression, *European Journal of Operational Research*, pp. 931–942. vol. 270, no.3.

[3] Bertsimas, D. & Gupta, V. & Kallus, N. (2017) Data Driven Robust Optimization, *Berlin Heidelberg and Optimization Society*, pp. 235–292. vol. 167, no.2.

[4] Bertsimas, D. & Dunn, J. (2019) *Machine Learning under a Modern Optimization Lens* Dynamic Ideas.

[5] Bertsimas, D. & Thiele, A. Robust and Data-Driven Optimization: Modern Decision-Making Under Uncertainty (2006).

[6]Ghaoui, Laurent. & Robust Solutions to Least-Squares Problems with Uncertain Data (1997), *SIAM Journal on Matrix Analysis and Applications*, vol. 18, no. 4, pp. 1035-1064.

[7] Tulabandhula, Theja. & Rudin, Cynthia (2014) Robust Optimization using Machine Learning for Uncertainty Sets.

[8] Artzner, P. & Delbaen, F. (1999) Coherent Measures of Risk. *Mathematical Finance*, 9:203-228.