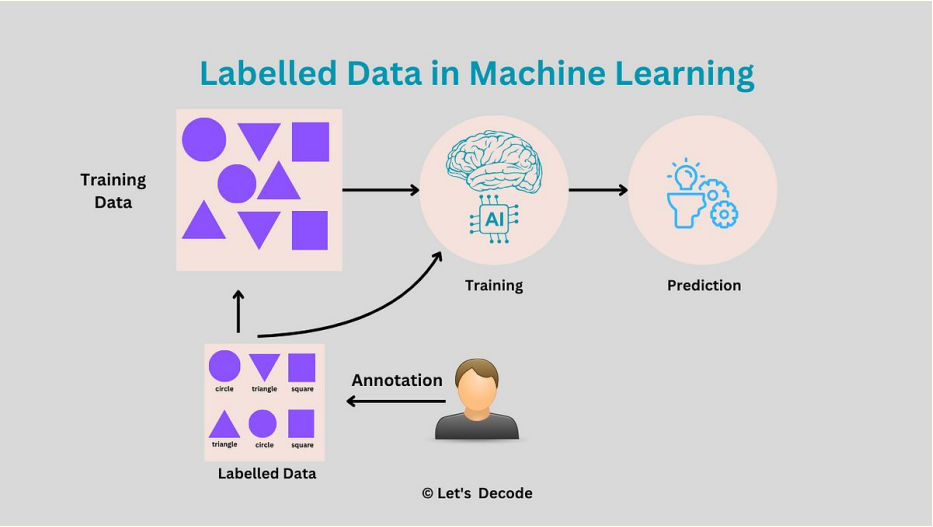
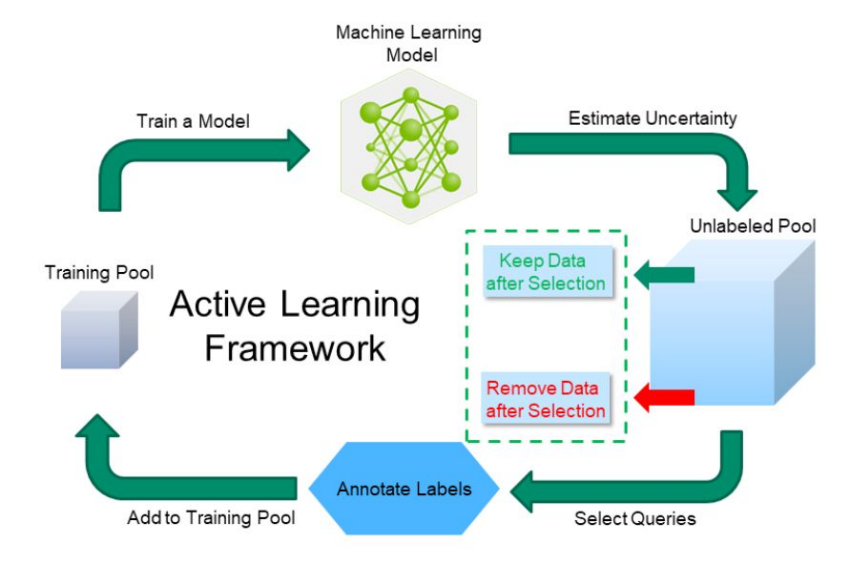
Troy Russo

Active Learning

The Problem


Labelled Data in Machine Learning
Training Data
AI
Training
Prediction
circle
triangle
square
triangle
circle
square
Labelled Data
Annotation
© Let's Decode

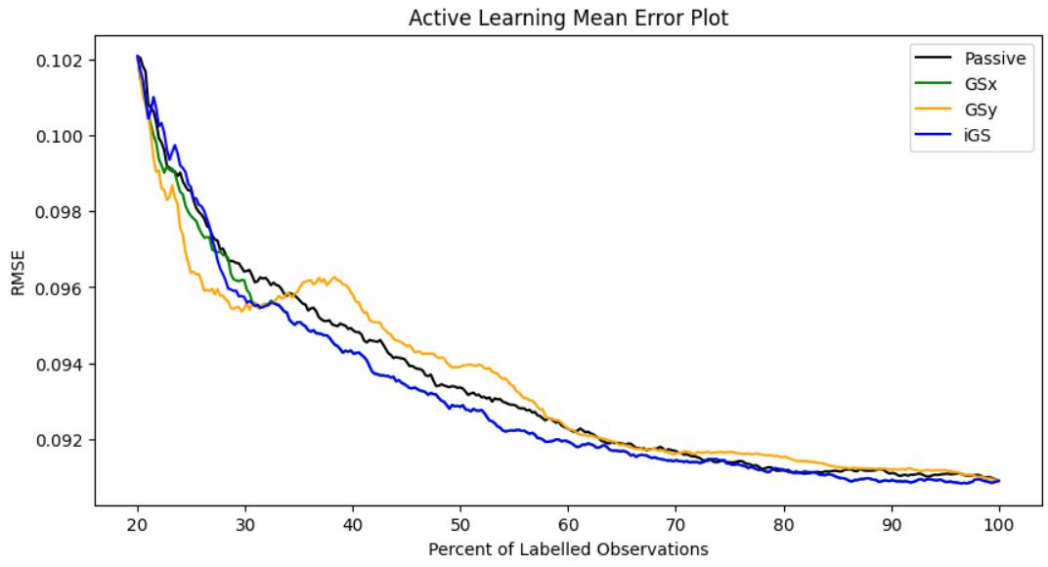
The Solution



- Start with unlabeled
- Label certain number of observations
- Run one iteration (linear regression or random forest common)
- Determine Sample(s) to label
- Repeat

When to stop

- RMSE stops decreasing significantly
- Not worth computational power
- Takes unneeded time to not improve model

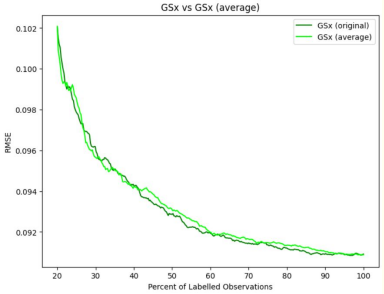


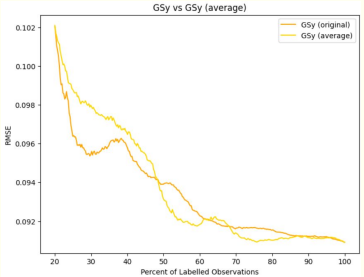
Lots of Different Methods

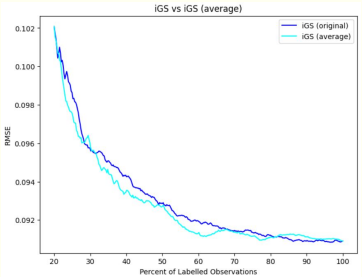
Uncertainty	Query By Committee	Diversity Sampling	Expected Model Change	Greedy Sampling
Select instances where the model's predictions are least confident	Use an ensemble of models and choose instances where there is disagreement among them	Ensure that the selected samples cover a wide range of scenarios, reducing redundancy	Pick samples that are expected to cause the largest change in the current model upon being labeled	Selects the single data point that appears most informative according to a specific criterion

- I am mainly concerned with Greedy Sampling
- Several Distinct types of Greedy Sampling (3 main)

Types of Greedy Sampling







GSx

- **Selects each sample by:**
  - **Finding shortest distance to each unlabeled sample (not average distance from each point)**
  - **Selecting farthest distance from this list**
- **Independent of regression model since it samples based on input (computationally cheaper than the next 2)**

GSy

- **Selects each sample by:**
  - **Finding shortest distance from every predicted value (based on regression) to the actual value**
  - **Selects farthest distance from this list**
- **Has to update regression model each time**

iGS

- **Selects each sample by:**
  - **Finding each distance matrix from GSx and GSy and multiplying these matrices**
  - **Selects farthest distance in this matrix**
- **Computationally similar to GSy since most of cost is updating regression**

Formulas and Variations

Formulas

GSx

$$d^{\mathbf{x}}_{nm} = ||\mathbf{x}_n - \mathbf{x}_m||, \quad m = 1, ..., k; n = k+1, ..., N$$

$$d^{\mathbf{x}}_{n} = \min_{m} d^{\mathbf{x}}_{nm}, \quad n = k+1, ..., N$$

GSy

$$d^{y}_{nm} = ||f(\mathbf{x}_n) - y_m||, \quad m = 1, ..., k; n = k+1, ..., N$$

$$d^{y}_{n} = \min_{m} d^{y}_{nm}, \quad n = k+1, ..., N$$

IGS

$$d^{\mathbf{x}y}_{n} = \min_{m} d^{\mathbf{x}}_{nm} d^{y}_{nm}, \quad n = k+1, ..., N$$

Standardization

- Instead of multiplying these distances we standardize
- Use z-dist to make x and y equal weight
- Add distances instead of multiplying

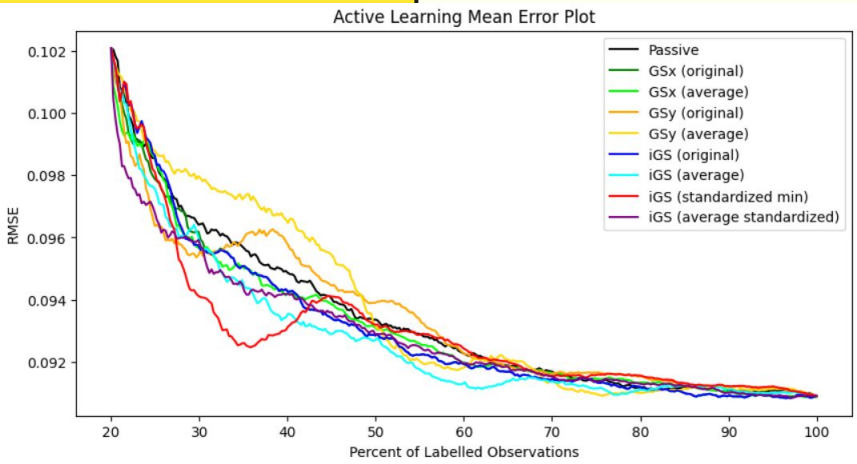
Average Distance

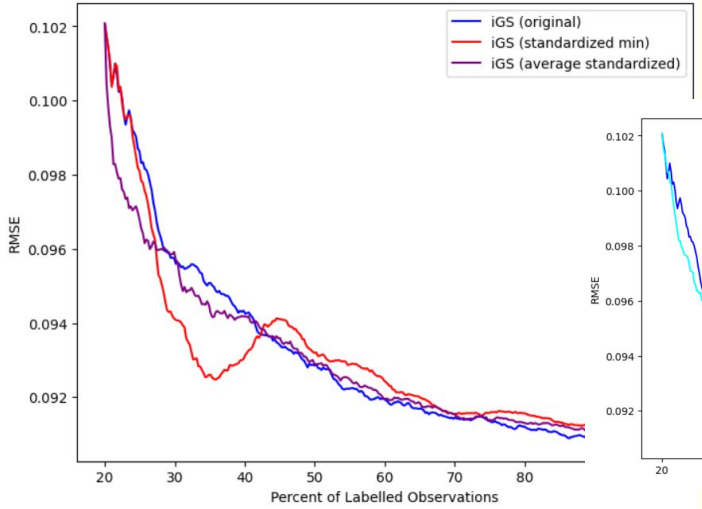
- Find average distance from each point
- Replaces min distance from each point

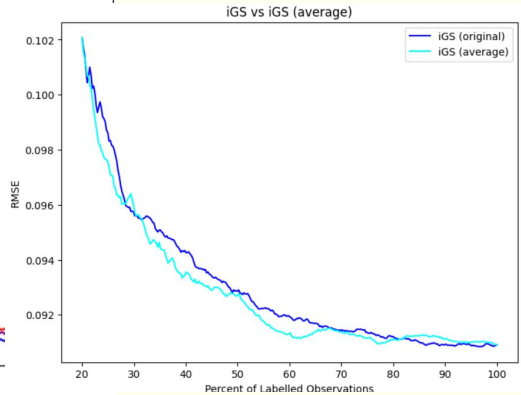
What's best?

- Large graph shows iGS variants are the best
- Significance tests show all iGS variants significantly different than GSx (cheaper)
- Significance tests/individual graphs show standardized iGS and iGS with average are the best two models.

	passive	GSx	GSxAvg	GSy	GSyAvg	iGS \
passive	1.000000e+00					
GSx	9.352560e-39	1.0				
GSxAvg	8.204135e-52	0.031426	1.0			
GSy	9.922432e-54	0.000008	0.116008	1.0		
GSyAvg	4.527727e-19	0.0	0.0	0.0	1.0	
iGS	9.730048e-41	0.001448	0.597028	0.000703	0.0	1.0
iGSAvg	5.554559e-35	0.0	0.0	0.0	0.000002	0.0
iGSStd	4.871225e-54	0.000772	0.034862	0.000015	0.0	0.030242
iGSAvgStd	5.761528e-52	0.300236	0.002289	0.009974	0.0	0.486444







Next Steps

- Use different machine learning models
  - Random forest, xgboost
- Train models on different datasets

Q&A

?