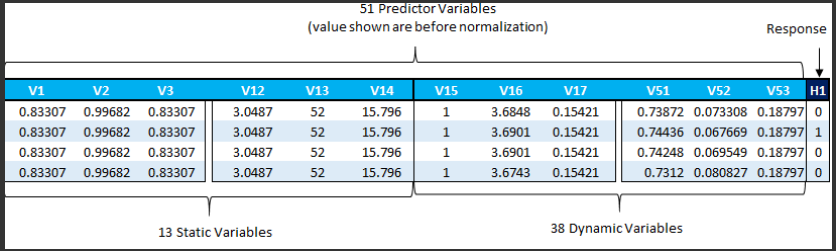
>>> Advanced Data Minining Project
>>> First-order Theorem Proving Data Set

Name: Anna Basanskaya
Date: December 19, 2017

>>> Table of Contents

>>> Data

* $p = 51^*$, $n = 6118$
* Used only the first of the five original response variables.
  * The response variable was converted from a censored quantitative variable to a binary variable.
  * The percentage of equal values among the original five response variables was 70.4%
* The percentage of 1s in the first response variable is 49.5%.

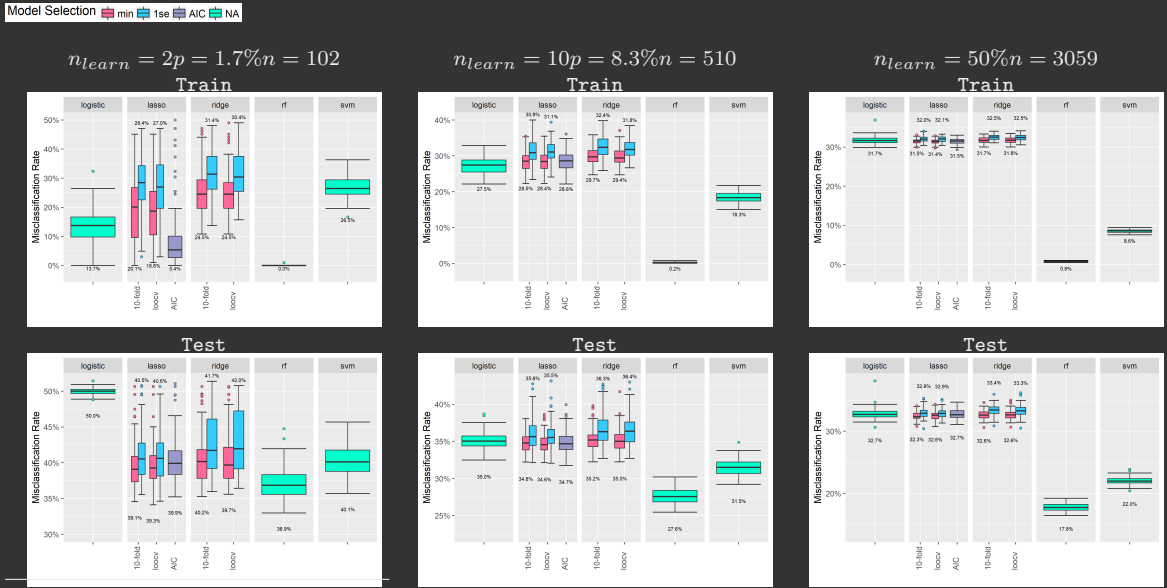


V1	V2	V3	V12	V13	V14	V15	V16	V17	V51	V52	V53	H1
0.83307	0.99682	0.83307	3.0487	52	15.796	1	3.6848	0.15421	0.73872	0.073308	0.18797	0
0.83307	0.99682	0.83307	3.0487	52	15.796	1	3.6901	0.15421	0.74436	0.067669	0.18797	1
0.83307	0.99682	0.83307	3.0487	52	15.796	1	3.6901	0.15421	0.74248	0.069549	0.18797	0
0.83307	0.99682	0.83307	3.0487	52	15.796	1	3.6743	0.15421	0.7312	0.080827	0.18797	0

* Variables (see the Appendix for details):
  * Static Predictors: derived from the description of the problem, e.g., fraction of clauses that are unit clauses.
  * Dynamic Predictors: measured using the proof state after the proof search started, e.g., proportion of generated clauses kept.
  * Response: Indicates whether a conjecture could be proved by Heuristic 1 within 100 seconds.
* Objective: Classify conjectures as proved/not proved within 100 seconds by Heuristic 1.

* After removing $V5$ and $V35$, which contained all zeros

>>> Boxplots†

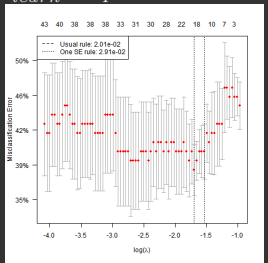


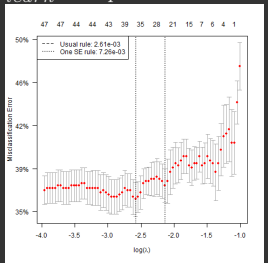
†Median values are shown in the charts. 1se values are always towards the top.

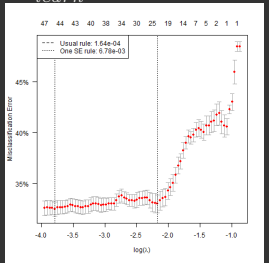
[2. Boxplots]$ _

$n_{learn} = 2p = 1.7\%n = 102$ | $n_{learn} = 10p = 8.3\%n = 510$ | $n_{learn} = 50\%n = 3059$

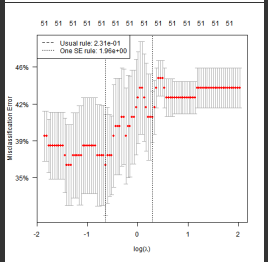
Lasso

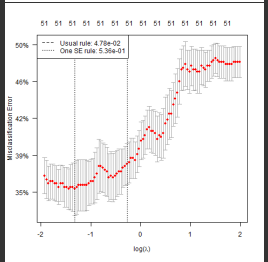


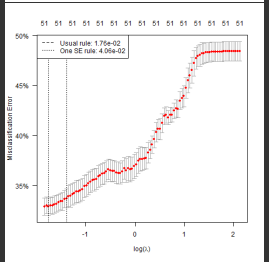




Ridge







⏭ Lasso 10-Fold CV and AIC

>>> AIC

- Background
  - KL Divergence measures the ``distance'' between the true distribution P and another distribution Q:

$$D_{kl}(P||Q) = \sum_i P(i) \log P(i) - P(i) \log Q(i).$$

  - When comparing estimated models and the true model is known, only the last term differs.
  - The true distribution is unknown in practice. Adding $2k$ results in an unbiased estimate of the KL divergence, leading to the definition

$$\text{AIC} = 2k - 2 \log \hat{L}$$

- One way to estimate AIC (up to a constant) is using glmnet outputs for k and $2 \log \hat{L} \equiv 2\hat{\mathscr{L}}$:
  - The deviance is defined to be 2*(loglike_sat - loglike), where loglike_sat is the log-likelihood for the saturated model (a model with a free parameter per observation).
  - Null deviance is defined to be 2*(loglike_sat - loglike(Null)). The NULL model refers to the intercept model, except for the Cox, where it is the 0 model.

$$\begin{aligned} D &= 2 \times (\hat{\mathscr{L}}_{\text{sat}} - \hat{\mathscr{L}}) && D \equiv \text{deviance},\ \hat{\mathscr{L}} \equiv \log \hat{L} \\ D_0 &= 2 \times (\hat{\mathscr{L}}_{\text{sat}} - \hat{\mathscr{L}}_0) && \text{sat denotes the saturated model} \\ D_0 - D &= 2 \times (\hat{\mathscr{L}} - \hat{\mathscr{L}}_0), && 0 \text{ denotes the null model} \end{aligned}$$

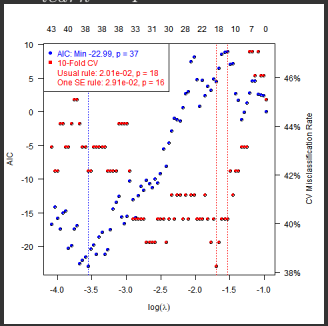
  - R code snippet:

```
LL.times.2 ← rep(glm.fits$nulldev, length(devs))-devs
k          ← glm.fits$df[(glm.fits$lambda %in% glmnet.result$lambda)]  #non-zero predictors for lasso
AIC        ← -LL.times.2 + 2*k
```

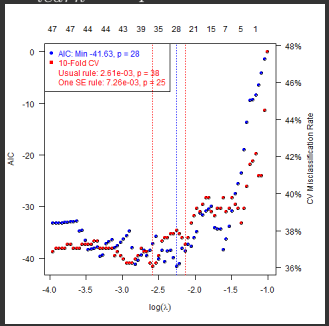
- Glmnet probably ignores constants since:
  - Regression (Guassian regression model†): $\log \hat{L} = -\frac{n}{2} \log 2\pi\hat{\sigma}^2 - \frac{1}{2\hat{\sigma}^2} \sum_{i=1}^{n} (y_i - x_i\hat{\beta}_i)^2$
  - The deviance ratio is R^2.
- The constants should not affect model comparison.

†Not used for this project since the objective was classification.

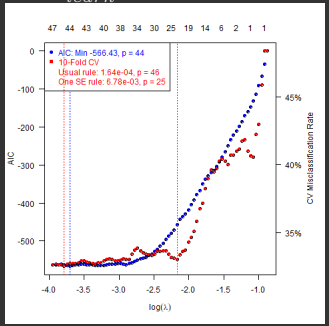
$n_{learn} = 2p = 1.7\%n = 102$



$n_{learn} = 10p = 8.3\%n = 510$



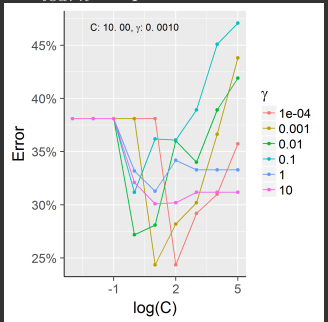
$n_{learn} = 50\%n = 3059$

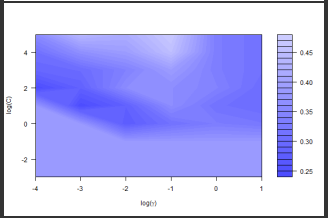


◂ Lasso and Ridge 10-Fold CV Curves

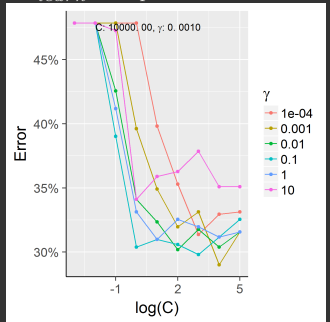
‡Up to a constant

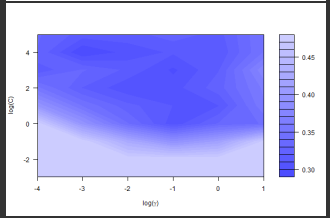
$n_{learn} = 2p = 1.7\%n = 102$



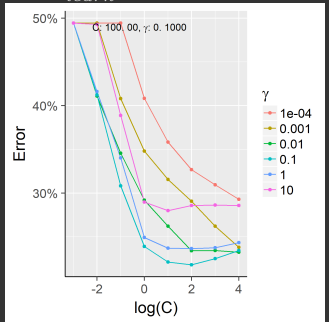


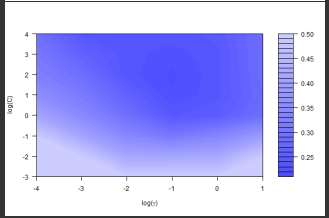
$n_{learn} = 10p = 8.3\%n = 510$



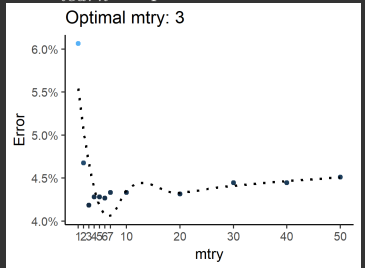


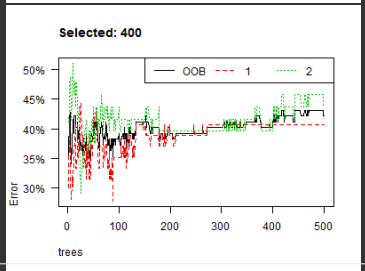
$n_{learn} = 50\%n = 3059$



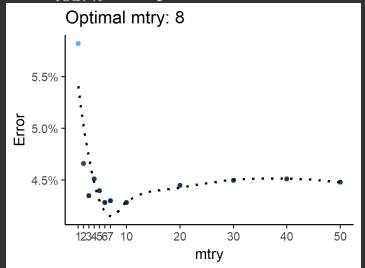


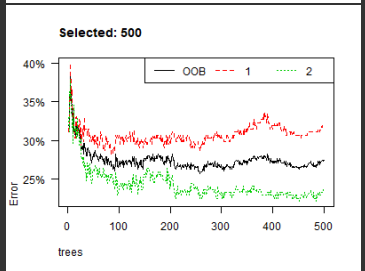
$n_{learn} = 2p = 1.7\%n = 102$



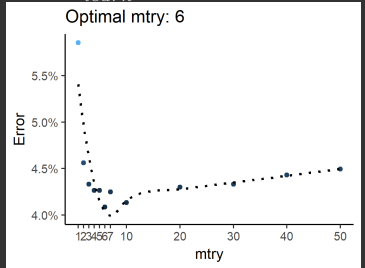


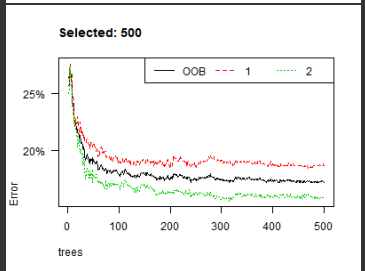
$n_{learn} = 10p = 8.3\%n = 510$





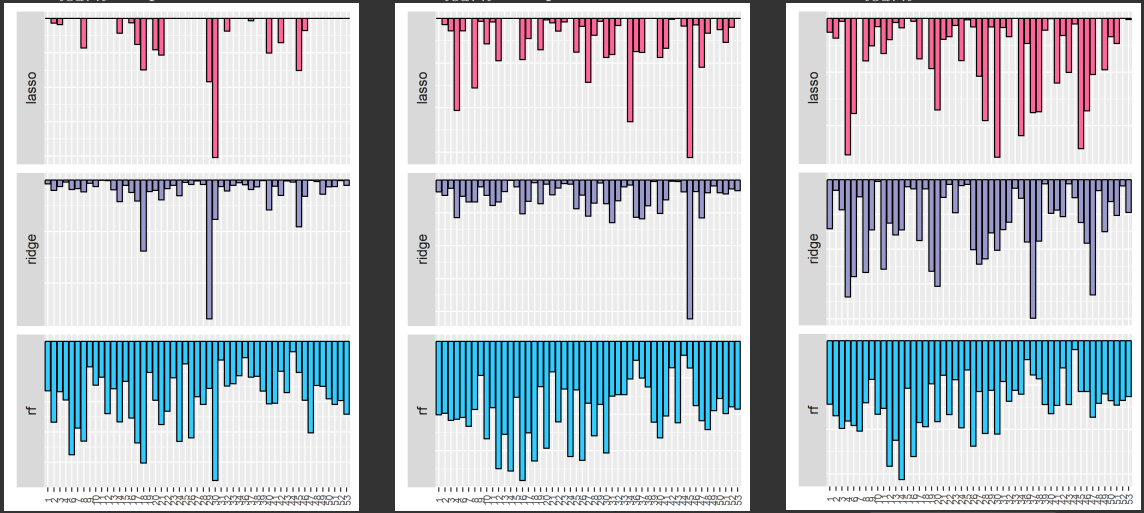
$n_{learn} = 50\%n = 3059$





§Ultimately, the default values were used since the values selected by playing around with the numbers were close to the defaults.

$n_{learn} = 2p = 1.7\%n = 102$ | $n_{learn} = 10p = 8.3\%n = 510$ | $n_{learn} = 50\%n = 3059$



◂ Lasso 10-Fold CV and AIC

¶For $\boldsymbol{2p}$, the absolute value of the coefficient for V19 is only 0.1% of the max and cannot be seen in the chart.

>>> Data Details

Variable	Description	Variable	Description	Variable	Description
V1	Fraction of clauses that are unit clauses.	V18	$\|U\|/\|A\|$	V35^	Ratio of the number of non-redundant deleted clauses to $\|P\|$.
V2	Fraction of clauses that are Horn clauses.	V19	Ratio of longest clause lengths in P and A.	V36	Ratio of the number of backward subsumed clauses to $\|P\|$.
V3	Fraction of clauses that are ground Clauses.	V20	Ratio of average clause lengths in P and A.	V37	Ratio of the number of backward rewritten clauses to $\|P\|$.
V4	Fraction of clauses that are demodulators.	V21	Ratio of longest clause lengths in U and A.	V38	Ratio of the number of backward rewritten literal clauses to $\|P\|$.
V5^	Fraction of clauses that are rewrite rules (oriented demodulators).	V22	Ratio of average clause lengths in U and A.	V39	Ratio of the number of generated clauses to $\|P\|$.
V6	Fraction of clauses that are purely positive.	V23	Ratio of maximum clause depths in P and A.	V40	Ratio of the number of generated literal clauses to $\|P\|$.
V7	Fraction of clauses that are purely negative.	V24	Ratio of average clause depths in P and A.	V41	Ratio of the number of generated non-trivial clauses to $\|P\|$.
V8	Fraction of clauses that are mixed positive and negative.	V25	Ratio of maximum clause depths in U and A.	V42	context_sr_count/$\|P\|$.
V9	Maximum clause length.	V26	Ratio of average clause depths in U and A.	V43	Ratio of paramodulations to $\|P\|$.
V10	Average clause length.	V27	Ratio of maximum clause standard weights in P and A.	V44	factor_count/$\|P\|$.
V11	Maximum clause depth.	V28	Ratio of average clause standard weights in P and A.	V45	resolv_count/$\|P\|$.
V12	Average clause depth.	V29	Ratio of maximum clause standard weights in U and A.	V46	Fraction of unit clauses in U.
V13	Maximum clause weight.	V30	Ratio of average clause standard weights in U and A.	V47	Fraction of Horn clauses in U.
V14	Average clause weight.	V31	Ratio of the number of trivial clauses to $\|P\|$.	V48	Fraction of ground clauses in U.
V15	Proportion of generated clauses kept. (Subsumed or trivial clauses are discarded.)	V32	Ratio of the number of forward subsumed clauses to $\|P\|$.	V49	Fraction of demodulator clauses in U.
V16	Sharing factor. (A measure of the number of shared terms.)	V33	Ratio of the number of non-trivial clauses to $\|P\|$.	V50	Fraction of rewrite rule clauses in $\|U\|$.
V17	$\|P\|/\|P \bigcup U\|$	V34	Ratio of the number of other redundant clauses to $\|P\|$.		

^Removed as all zeros

- The E automatic prover was used.
- The set of processed clauses is denoted by P and the set of unprocessed clauses by U. The set of axioms is denoted by A. context_sr_count, factor_count and resolv_count are variables within E.
- Conjectures were taken from Problems for Theorem Provers (TPTP).
- Heuristic 1: G_E_021_K31_F1_PI_AE_S4_CS_SP_S2S as labeled by E; e.g., _PI denotes a preference for initial clauses, _SP denotes simultaneous paramodulation. Source: https://archive.ics.uci.edu/ml/datasets/First-order+theorem+proving

◂ Data slide