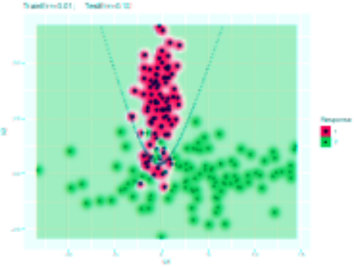
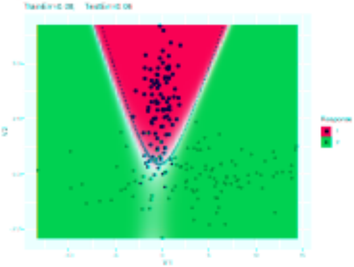
RECAP: OVERFITTING

- Occurs when model reflects noise or artifacts in training data
- Model often then does not generalize well (small train error, high test error) – or at least works better on train than on test data

Overfitted model

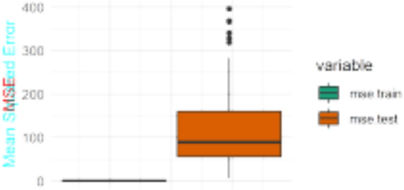


Appropriate model



EXAMPLE I: OVERFITTING

- Data set: daily maximum **ozone level** in LA; $n = 50$
- 12 features: time (weekday, month); weather (temperature at stations, humidity, wind speed); pressure gradient
- Orig. data was subsetted, so it feels “high-dim.” now (low n in relation to p)
- LM with all features (L2 loss)
- MSE evaluation under 10×10 REP-CV



Model fits train data well, but generalizes poorly.

EXAMPLE II: OVERFITTING

- We train an MLP and a CART on the mtcars data
- Both models are not regularized
- And configured to make overfitting more likely

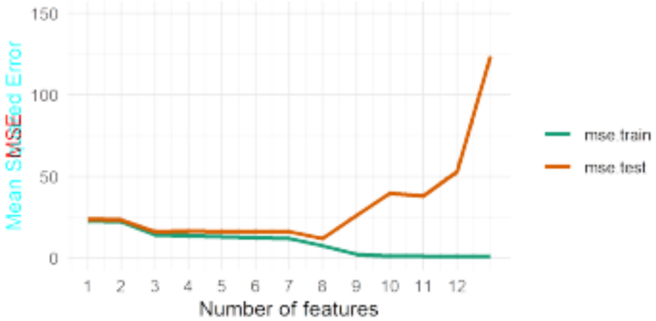
	Train MSE	Test MSE
Neural Network	1.47	345.84
CART	0.00	16.91

(And we now switch back to the Ozone example...)

AVOIDING OVERFITTING – REDUCE COMPLEXITY

We try the simplest model: a constant. So for $L2$ loss the mean of $y^{(i)}$.

We then increase complexity by adding one feature at a time.



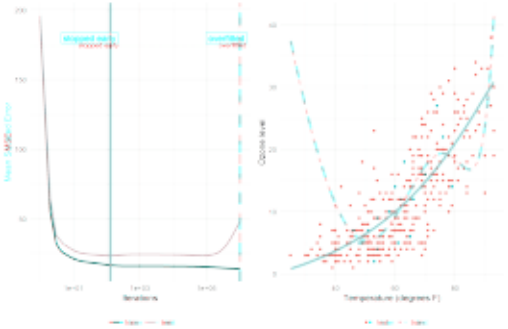
NB: We added features in a specific (clever) order, so we cheated a bit.

AVOIDING OVERFITTING – OPTIMIZE LESS

Now: polynomial regression with temperature as single feature

$$f(\mathbf{x} \mid \boldsymbol{\theta}) = \sum_{k=0}^{d} \theta_k \cdot (x_T)^k$$

We set $d = 15$ to overfit to small data. To investigate early stopping, we don't analytically solve the OLS problem, but run GD stepwise.



We see: Early stopping GD can improve results.

NB: GD for poly-regr usually needs many iters before it starts to overfit, so we used a very small training set.