Generalization Error Bounds for State Space Models

with an application to economic forecasting

Thesis Proposal

Daniel McDonald

Committee:

Cosma Shalizi, Mark Schervish, Alessandro Rinaldo,
Larry Wasserman, and David N. DeJong

July 22, 2010

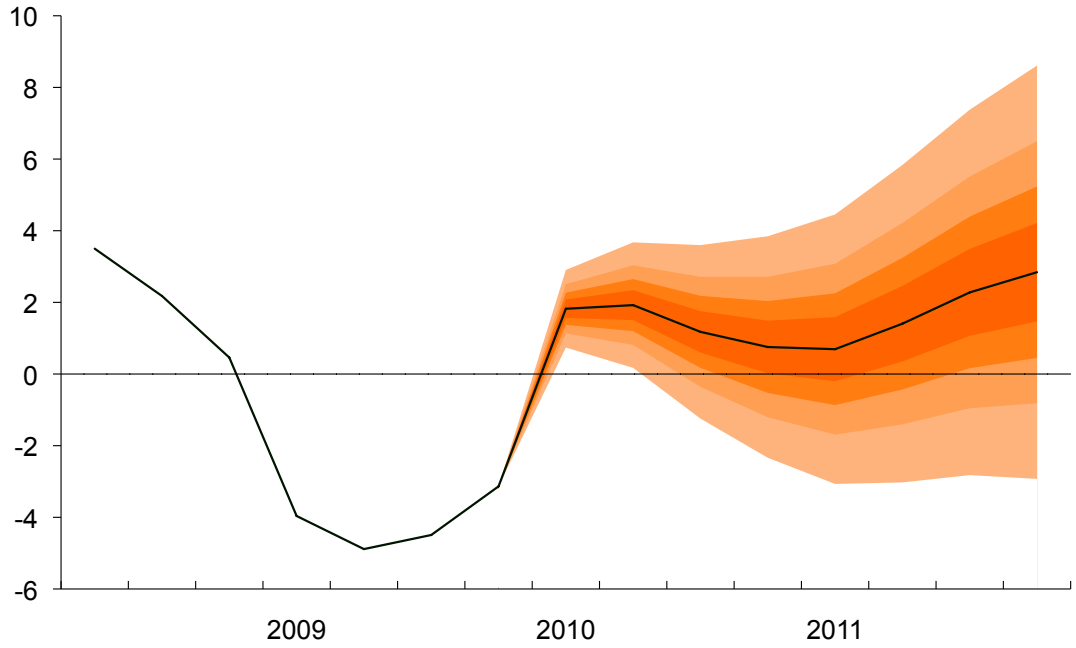
FORECASTING

- Given some data

$$x_1, \ldots, x_T \in \mathcal{X}$$

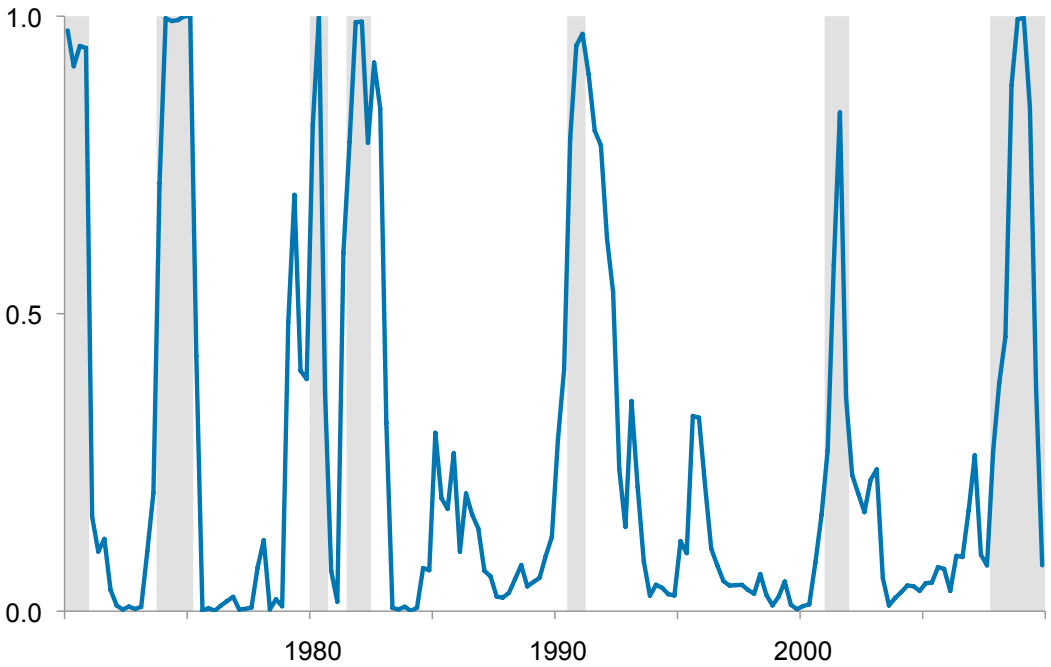
- Want to predict the next data point(s)

$$x_{T+1}, \ldots, x_{T+k}$$



Source: Czech National Bank

Methods of economic forecasting

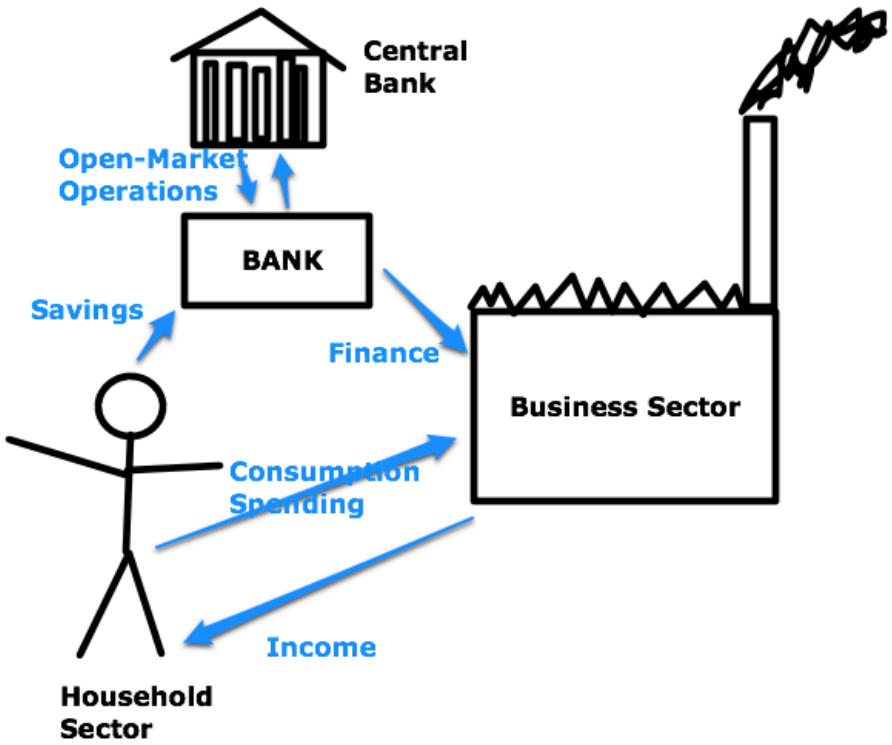


- ARIMA, ARFIMA, GARCH, etc.
- Dynamic Factor Models (Hamilton, Chib, Kim and Nelson, others)
- Systems of Equations models
- Dynamic Stochastic General Equilibrium (DSGE) models

Source: Econbrowser Recession Probabilities

DSGE MODELS

- Most active area of macroeconomic research in the last 30 years
- Arose in response to the Lucas (1976) critique
- Pioneered by Kydland and Prescott (1982)
- Attempt to incorporate "rational behavior" into forecasting models
- Have come under fire for being unable to forecast the financial collapse of 2008–?



Source: Brad DeLong's realization of Daniel Davies' DSGE model

RBC MODEL

- Imagine an infinitely long lived individual who faces the following constrained optimization problem:

$$\begin{aligned}
\max_{c_t, l_t} U &= \mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t u(c_t, l_t) \\
y_t &= z_t q(k_t, n_t) \\
1 &= n_t + l_t \\
y_t &= c_t + i_t \\
k_{t+1} &= i_t + (1 - \delta) k_t \\
z_t &\sim \text{AR}(1)
\end{aligned}$$

Relationship to state space models

DSGE Model

$$\max_{c_t, l_t} U = \mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t u(c_t, l_t)$$

$$y_t = z_t q(k_t, n_t)$$

$$1 = n_t + l_t$$

$$y_t = c_t + i_t$$

$$k_{t+1} = i_t + (1 - \delta) k_t$$

$$z_t \sim \text{AR}(1)$$

Math games

State Space Model

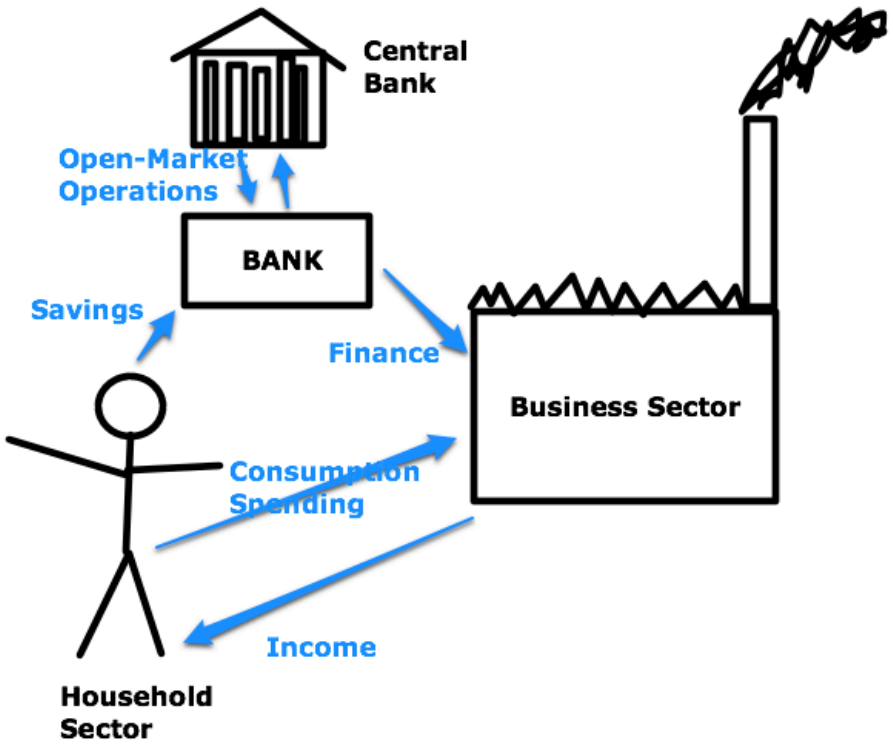
$$\mathbf{x}_t = g(\boldsymbol{\alpha}_t, \epsilon_t)$$

$$\boldsymbol{\alpha}_{t+1} = h(\boldsymbol{\alpha}_t, \eta_{t+1})$$

$$\boldsymbol{\alpha}_1 \sim F$$

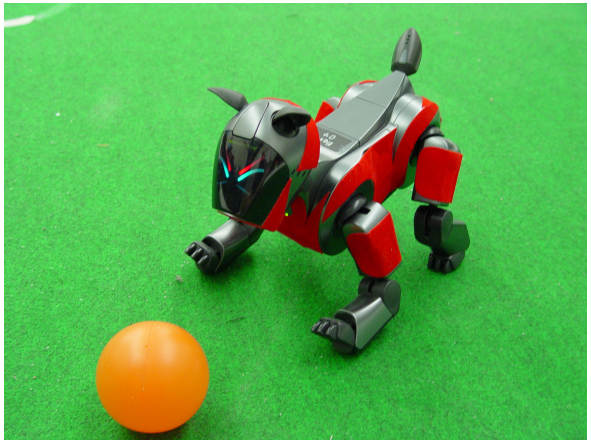
Simplicity/Complexity

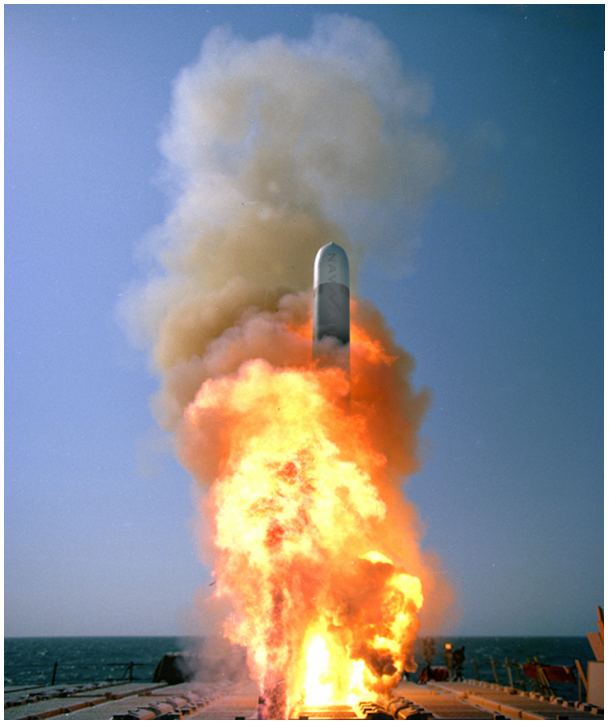
- Unclear if this model is "good"
- Lots of economic arguments Pro/Con
- What about statistical behavior?
- Overfit/Underfit
- How do predictions compare to other SS models?

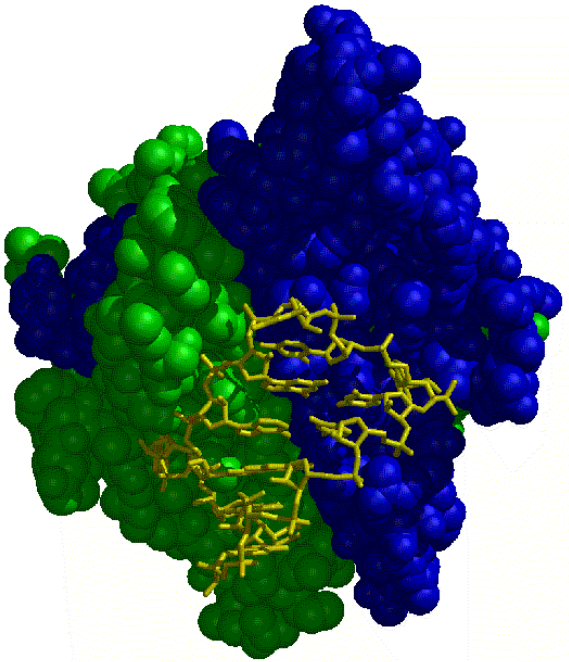


State space models

- Lots of disciplines use state space models
- Sometimes motivated directly by physical relationships
- How can we measure the forecasting performance?







Thesis project

Proposal

Develop probabilistic bounds on the prediction error of state space models.

Plan of talk

Forecasting framework

1. Observe training data $D_n = \{(Y_1, X_1), (Y_2, X_2), \ldots, (Y_n, X_n)\}$ from some stochastic process μ
2. Choose model class $\mathcal{F}$ from which to construct predictors, e.g. AR(p), DSGE, regression, wavelets, etc.
3. Use a loss function $\ell(Y, f(X))$ to measure performance of candidate predictors $f \in \mathcal{F}$
4. Estimate the model using D_n, to produce $\widehat{f}$, your proposed forecasting model

Generalization error

- Want to control the generalization error, or risk, of chosen predictor $\widehat{f}$

$$R(\widehat{f}) = \mathbb{E}_{\mu}[\ell(Y_0, \widehat{f}(X_0)) \mid D_n]$$

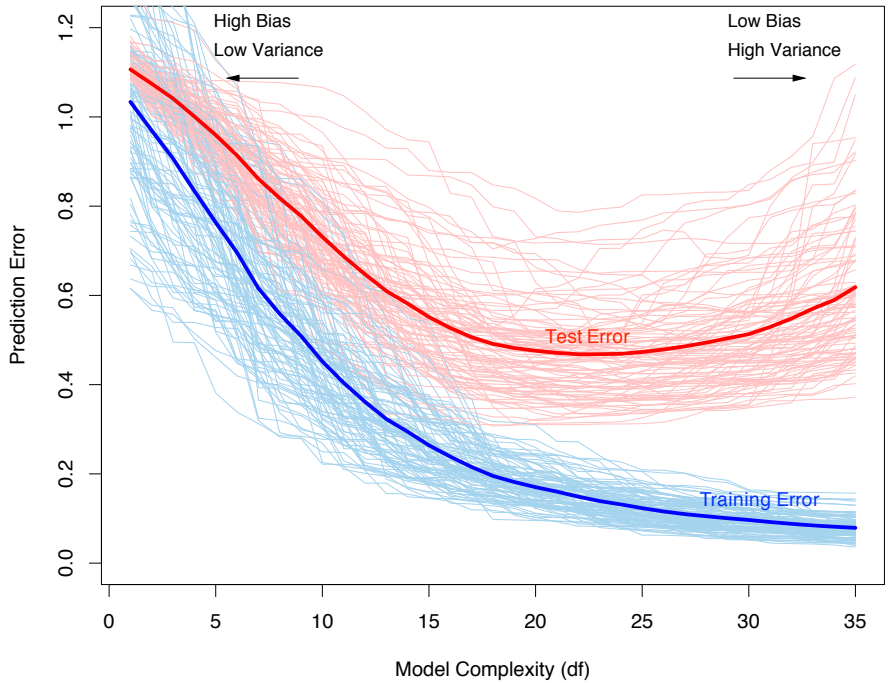
- But the stochastic process μ is unknown
- Usually estimate $R(\widehat{f})$ with training error

$$R_n(\widehat{f}) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, \widehat{f}(X_i))$$

- This is a poor estimate of $R(\widehat{f})$

Training error

- Often $R_n(\widehat{f}) < R(\widehat{f})$
- Complicated classes $\mathcal{F}$ lead to small $R_n(\widehat{f})$
- Simple classes $\mathcal{F}$ lead to large $R_n(\widehat{f})$
- Model comparisons using $R_n(\widehat{f})$ lead to choosing overly complex $\mathcal{F}$—overfitting
- Related to the bias/variance tradeoff



Source: Hastie, Tibshirani, and Friedman *The Elements of Statistical Learning*

Error bounds

Risk

$$R(\widehat{f}) = \mathbb{E}_{\mu}[\ell(Y_0, \widehat{f}(X_0)) \mid D_n]$$

- Estimation of $R(\widehat{f})$ is a hard problem since μ is unknown
- Instead, derive probabilistic upper bounds
- These bounds depend on $\mathcal{F}$—one needs to characterize the complexity of different function classes, as well as the algorithms used to choose $\widehat{f}$
- Many complexity measures—VC dimension, covering numbers, algorithmic stability, and Rademacher complexity

Rademacher complexity

Definition

Define the Rademacher complexity of a function class $\mathcal{F}$ as

$$\mathfrak{R}(\mathcal{F}) = \mathbb{E}_X \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{F}} \left| \frac{2}{n} \sum_{i=1}^{n} \sigma_i f(x_i) \right| \right],$$

where σ_i are iid and $\mathbb{P}(\sigma_i = 1) = \mathbb{P}(\sigma_i = -1) = \frac{1}{2}$.

- Measures the maximum covariance between the predictions and random noise—how closely can some $f \in \mathcal{F}$ fit garbage?
- Gives tight bounds
- Removing $\mathbb{E}_X$ gives empirical Rademacher complexity
- Somewhat ugly, hard to calculate

IID RESULTS FOR DEPENDENT DATA

- IID bounds require n independent training samples
- Applying IID results to dependent data requires knowing the number of "independent" samples ($< n$)

INTUITION

For dependent data, knowledge of a few data points gives information about other data points. The marginal benefit of observing new data is therefore less than if the data were independent

- Measures of mixing "adjust" the IID bounds

Mixing

Definition

Let $\{X_i\}_{i=1}^{\infty}$ be random variables generated from a stationary process. Let $\mathbb{P}_{t \otimes t+m}$ denote the joint distribution between $\{X_i\}_{i=1}^{t}$ and $\{X_i\}_{i=t+m}^{\infty}$ with $\mathbb{P}_t$ and $\mathbb{P}_{t+m}$ the associated marginals. Then the β-mixing coefficient is given by

$$\beta(m) = \sup_t ||\mathbb{P}_t \otimes \mathbb{P}_{t+m} - \mathbb{P}_{t \otimes t+m}||_{TV}.$$

- Sufficient condition[1] to apply IID results to dependent data is

$$\beta(m) \xrightarrow{m \to \infty} 0$$

- If this holds, the process is β-mixing

[1] Yu 1994

Time series bounds

Theorem

Let $\mathcal{H}$ be the space of losses bounded above by M. Then given a sample from a stationary β-mixing distribution, for all $m, a > 0$ with $2ma = n$ and $\eta > 2(a-1)\beta(m)$, then for all $f \in \mathcal{F}$, with probability at least $1 - \eta$,

$$R(f) < R_a(f) + \mathfrak{R}_a(\mathcal{H}) + M\sqrt{\frac{\log 2/\eta'}{2a}}$$

with $\eta' = \eta - 2(a-1)\beta(m)$.

Source: Mohri and Rostamizadeh 2009

Implications

Theorem

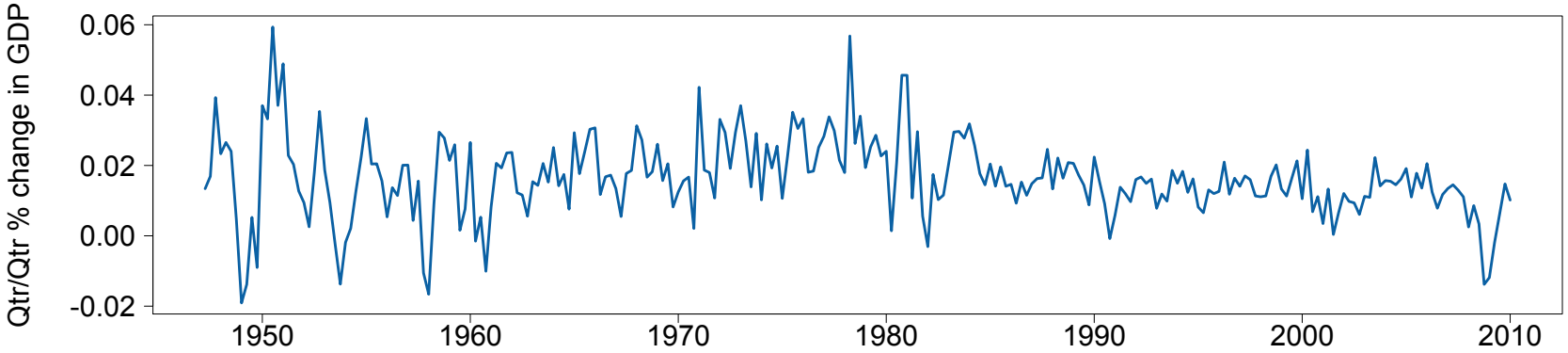
$$R(f) < R_a(f) + \mathfrak{R}_a(\mathcal{H}) + M\sqrt{\frac{\log 2/\eta'}{2a}}$$

- The effective sample size is not n but a
- The empirical risk is based on a data points separated by a distance $2m$
- Faster decay in $\beta(m)$ means more 'independent' samples, smaller third term
- Second term is Rademacher complexity of the loss space
- Can substitute empirical Rademacher complexity with slight modifications
- M is an upper bound for the loss

This bound does not work!

- Do not yet know Rademacher complexity for general state space models
- The β-mixing rate is assumed known—the bound depend of properties of unknown data generating process
- The bounded loss requirement is unpleasant for regression problems

Example



1. GDP data from 1947–2010 $(n = 252)$
2. Fit an AR(2) model
3. Assuming it is true, calculate β mixing rate
4. Calculate empirical Rademacher complexity (uses properties of regularized kernel regression)
5. Use squared error loss, truncated at M

With probability at least 0.85,

$$R(\widehat{f}) \leq 1 \times 10^{-4} + 0.07\sqrt{M} + 1.03M$$

What happened?

- For all M, $M <$ bound
  - ➥ Bound holds, but it is trivial
- 0.85 confidence level is strange
  - ➥ Due to $\beta(m)$
- There is not much data
  - ➥ $\sim$ 14 'independent' data points!
- Second and third terms have the wrong order of magnitude

Components of project

Proposal

Develop generalization error bounds for state space models to control the predictive risk and allow for model selection.

1. Estimate mixing behavior from data, adjust to accomodate extra uncertainty
2. Measure complexity of state space models
3. Modify bounds to allow for unbounded loss
4. Investigate resampling methods

Estimation of mixing rates

Three avenues for estimation so far:

1. 'Plug-in' method
   - Estimate the two densities using standard techniques
   - Numerical integration to get the total variation distance
   - Badly biased (upward)
2. 'Information' method
   - Can bound β-mixing: $\beta(m) \leq \sqrt{I(m)}$
   - $I(m)$ is just mutual information
   - Bound may be loose, some methods have very bad behavior for nearly independent distributions
3. 'Copula' method
   - Total variation distance is invariant under 1-1 transformations
   - Problem reduces to estimating the copula and a simpler numerical integration

When solving a given problem, try to avoid solving a more general problem as an intermediate step. —Vladimir Vapnik

Complexity characterization

1. Rademacher complexity
   - Gives tight bounds
   - Can calculate for stationary AR (regularized kernel problem)
   - Not so obvious for more complicated models (VAR, linear SS, nonlinear SS)
2. VC-dimension
   - More likely to work for complicated models—Can likely calculate simply
   - Available bounds not in terms of VC-dim

Unbounded loss

- Requires altering the existing bounds
- Need different concentration inequalities
- These require different assumptions on the data generating process
  - Result from Jiang (2010)—Need to control

$$\int_c^\infty |X_t| dp(x)$$

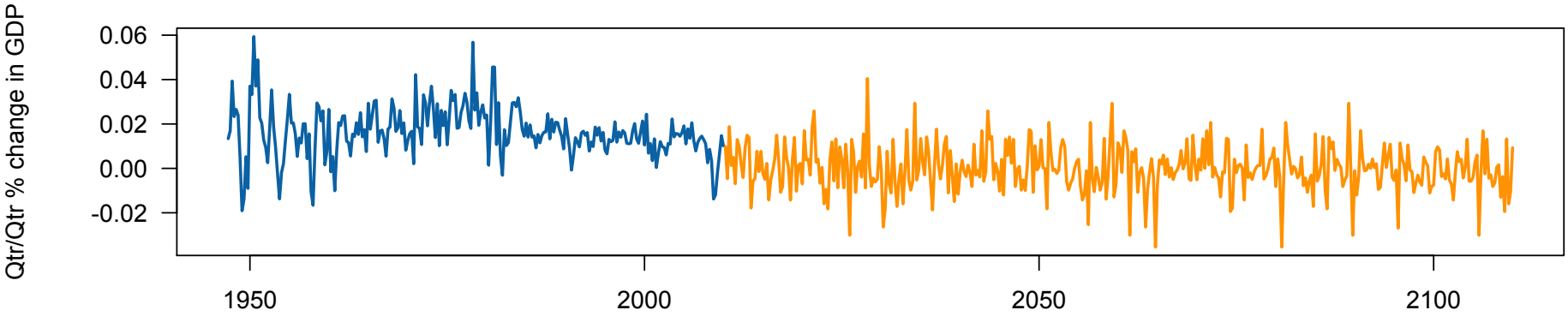
  - Result from Vapnik—Need to control

$$\sup_{f \in \mathcal{F}} \frac{\left(\int \ell^p(Y_0, f(X_0)) d\mu \right)^{1/p}}{\int \ell(Y_0, f(X_0)) d\mu}$$

Alternative methods

1. Cross-validation/Subset prediction
   - Most common 'validation' method in economic forecasting literature
   - Some theory
   - Models are complex/not much data
   - Models are slow to estimate
   - Modelers have observed held out data thoroughly
2. Bootstrapping
   - Need theory for time series generalization error
   - Models are slow to estimate
   - Could work...

Bootstrap error bound



1. Fit an AR(2) model and calculate the training error
2. Repeat B times:
   - Bootstrap a new series which is several times longer than the original
   - Fit a model to initial portion and calculate the in-sample risk
   - Calculate the prediction risk on the remainder
   - Store the difference between the in-sample and prediction risks
3. Find the 95% and 85% of the distribution of over-fits. Add this to the training error

$$R(\widehat{f}) \leq 1.5 \times 10^{-4} \qquad R(\widehat{f}) \leq 1.35 \times 10^{-4}$$

Conclusions

Proposal

Develop generalization error bounds for state space models to control the predictive risk and allow for model selection.

1. Estimate mixing behavior from data, adjust the bound(s) to accomodate this additional uncertainty
2. Measure complexity of state space models
3. Modify bounds to allow for unbounded loss, refine bounded loss results (bounded bounds!)
4. Investigate resampling methods

The End

Thanks for coming

Stationary AR

- Bound the Rademacher complexity of the class of models

$$\mathcal{F}_p = \left\{ \varphi_1, \ldots, \varphi_p : x_t = \sum_{i=1}^{p} \varphi_i x_{t-i} \text{ and } x_t \text{ is stationary} \right\}$$

- Stationarity requires the roots of $p(z) = z^p + \varphi_1 z^{p-1} + \cdots + \varphi_p$ lie inside the complex unit disc.
- Can show that a sufficient condition is[1]

$$||\varphi||_2^2 \leq \sum_{i=1}^{p} \binom{p}{i}^2 = \binom{2p}{p} - 1$$

- Bounds the norm

[1] Fam and Meditch 1978

Stationary AR

- Ordinary linear regressions can be written as kernel regressions. Let

$$\alpha_i = \left(\mathbf{X}(\mathbf{X}'\mathbf{X})^{-2}\mathbf{X}'\mathbf{Y}\right)_i$$
$$k(\mathbf{X}_i, \mathbf{X}_j) = \mathbf{X}_i\mathbf{X}_j',$$

  where $\mathbf{X}$ is the $n \times p$ design matrix, $\mathbf{Y}$ are the responses, and $\mathbf{X}_i$ is the i^{th} row of the design matrix.
- Requiring $\sum_{i,j} \alpha_i \alpha_j k(\mathbf{X}_i, \mathbf{X}_j) \leq \gamma^2$
- Corresponds $||\widehat{\beta}^{OLS}||_2^2 \leq \gamma^2$, or ridge regression

Stationary AR

$$\mathcal{F}_p \subseteq \overline{\mathcal{F}_p} = \left\{ \varphi_1, \ldots, \varphi_p : x_t = \sum_{i=1}^{p} \varphi_i x_{t-i} \text{ and } ||\varphi||_2^2 \leq \binom{2p}{p} - 1 \right\}$$

Allows application of kernel regularized result[1]

$$\mathfrak{R}(\mathcal{F}_p) \leq \mathfrak{R}(\overline{\mathcal{F}_p}) \leq \frac{2}{\sqrt{n}} \sqrt{\left(\binom{2p}{p} - 1 \right) \mathbb{E}\mathbf{X_1}\mathbf{X_1}'}$$

$$\mathfrak{R}_n(\mathcal{F}_p) \leq \mathfrak{R}_n(\overline{\mathcal{F}_p}) \leq \frac{2}{\sqrt{n}} \sqrt{\left(\binom{2p}{p} - 1 \right) \frac{1}{n} \sum_{t=i}^{n} \mathbf{X_i}\mathbf{X_i}'}$$

[1] Bartlett and Mendelson 2002

Example

Theorem

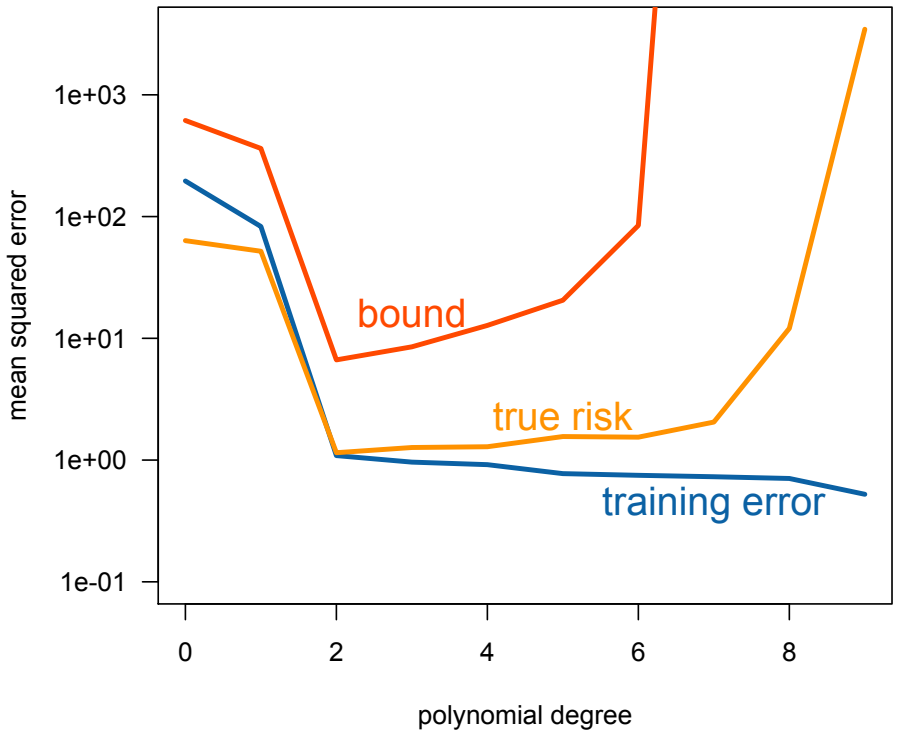
Let $h = d_{\mathcal{F}}/n$ where $d_{\mathcal{F}}$ is the VC dimension of $\mathcal{F}$. Then with probability at least $1 - \eta$

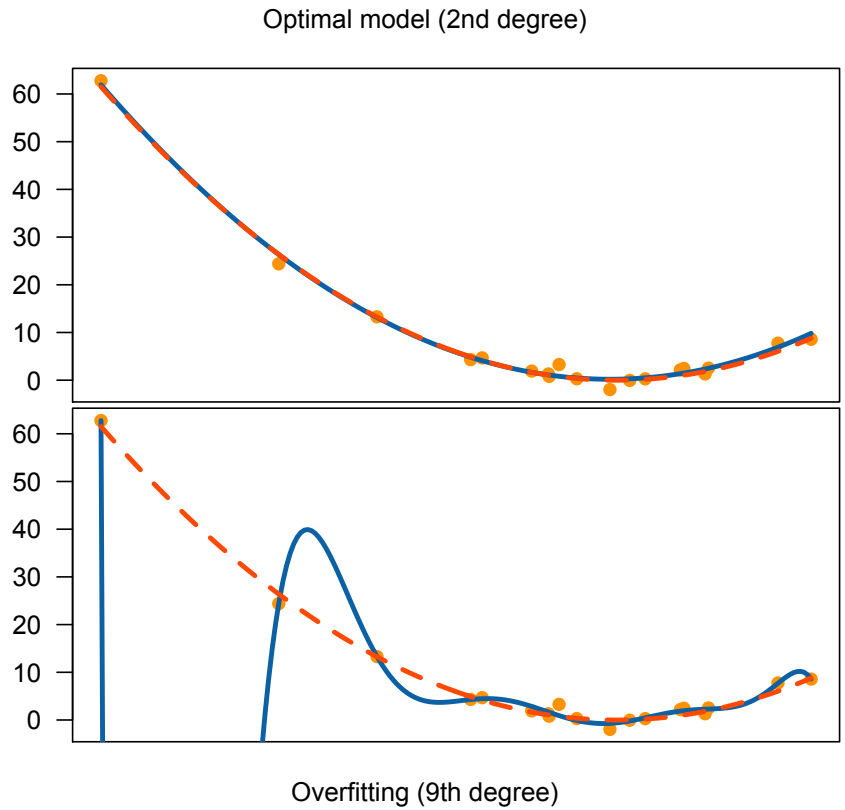
$$R(\widehat{f}) \leq R_n(\widehat{f}) \times \left(1 - \sqrt{h - h\ln h - \frac{\ln \eta}{n}}\right)_+^{-1}$$

- Applies to all nonnegative loss functions ℓ and distributions μ satisfying certain conditions[1]
- Is easy to apply for many classes $\mathcal{F}$
- Can use for model selection

[1] Vapnik

Example


mean squared error
1e+03
1e+02
1e+01
1e+00
1e-01
0
2
4
6
8
polynomial degree
bound
true risk
training error


Optimal model (2nd degree)
60
50
40
30
20
10
0
Overfitting (9th degree)

On-line learning results

Cesa-Bianchi, Conconi, and Gentile 2004

- Nice results about generalization error (rather than regret) in on-line framework
- Main idea—Take the hypothesis generated at the end of each time step, calculate the loss on future data, add complexity penalty. Hypothesis with smallest penalized empirical risk is used. Its risk is bounded easily.
- Suffers from a few issues
  1. Applies to IID data
  2. Bounded loss
  3. Bootstrap/CV flavor is computationally difficult

We close by mentioning an important open question: Is it possible to extend the results...to the case when the examples are not independent; e.g., when they are generated by a stationary process?

Could be a nice problem to tackle along the way

On-line learning results

Rakhlin, Sridharan, and Tewari 2010

- Main idea—Provides conditions which ensure online learnability
- The online learning problem is cast as a game played on rooted binary trees
- Defines complexity measures for bounding the value of the game (regret) of an online learner
- Important result for time series—Finite fat-shattering dimension implies universal uniform convergence (an analogue of Glivenko-Cantelli for dependent data)
- Bounded function classes with finite fat shattering dimension are online learnable
- In this case, various complexity measures (including the authors' sequential Rademacher complexity) are within $O(\log^{3/2} n)$

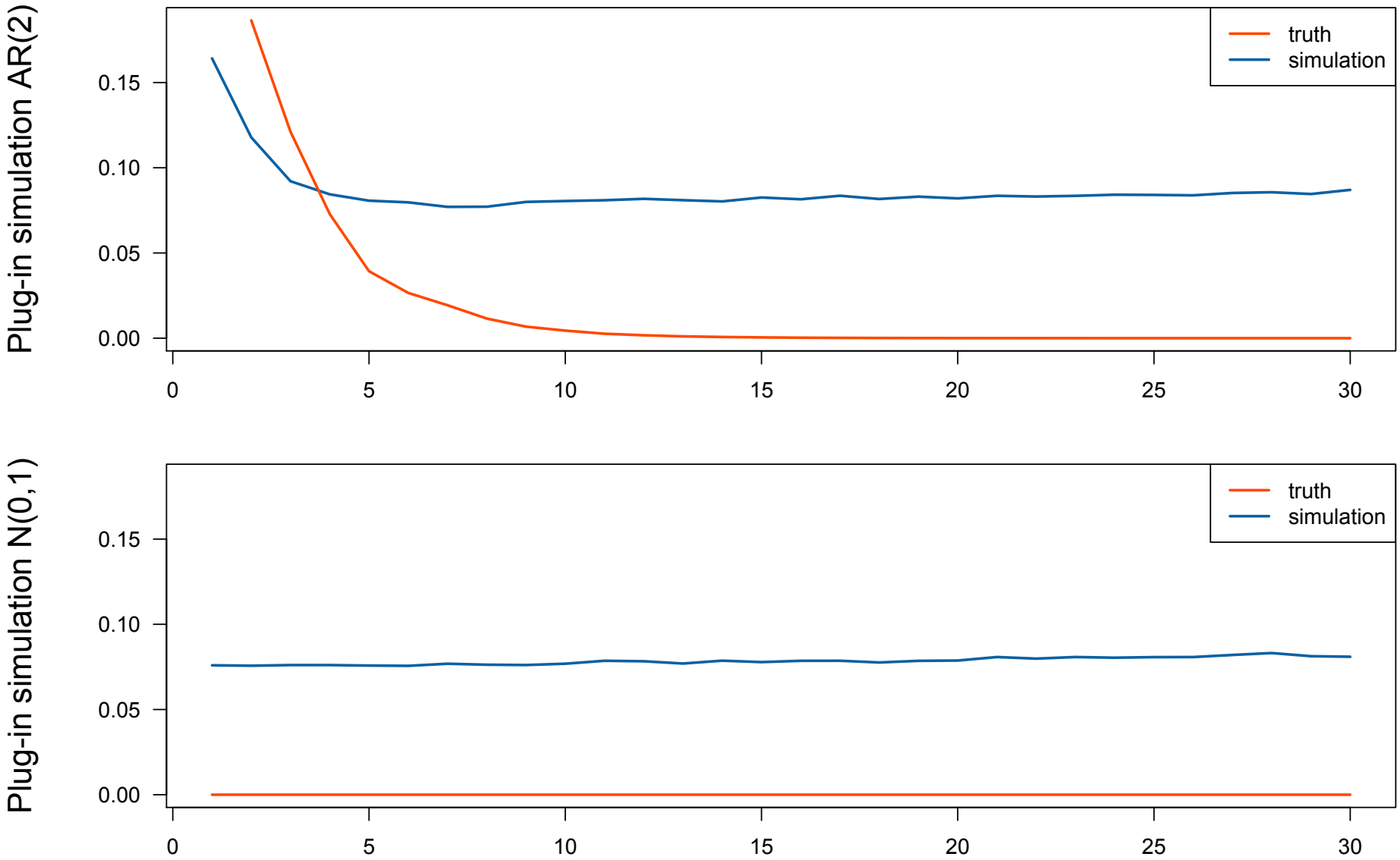
Estimating β-mixing

We want to estimate

$$\begin{aligned}\beta(m) &= \sup_t ||\mathbb{P}_t \otimes \mathbb{P}_{t+m} - \mathbb{P}_{t\otimes t+m}||_{TV} \\ &= \frac{1}{2}\int\int |p_t(x) \otimes p_{t+m}(y) - p_{t\otimes t+m}(x,y)| dxdy\end{aligned}$$

- Since we have assumed stationarity $p_t(x) = p_{t+m}(y)$
- Thus there is only a univariate density and a bivariate density to estimate

Estimating β-mixing—results



Estimating information mixing

We want to estimate

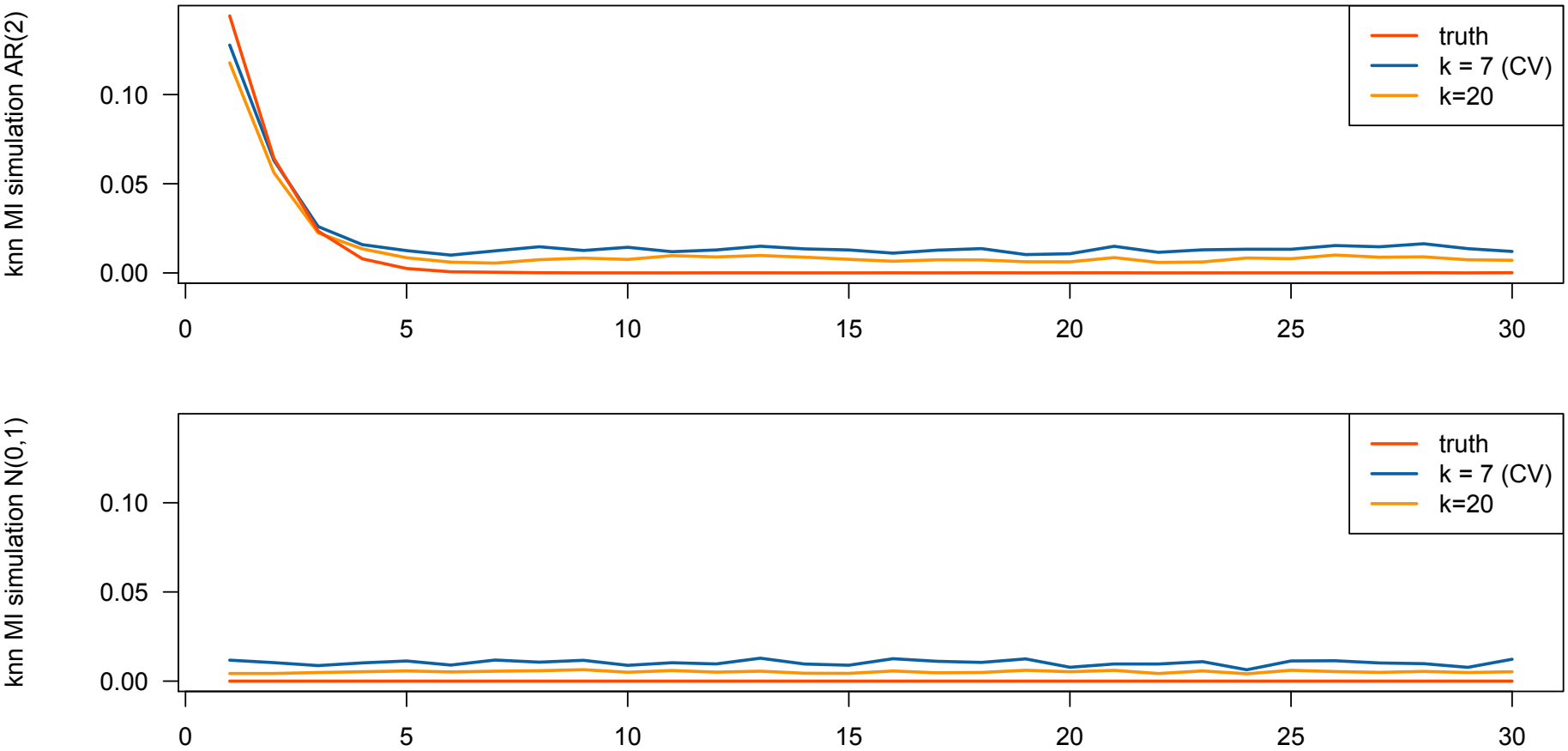
$$I(m) = \int \log \frac{p_{t \otimes t+m}(x,y)}{p_t(x) p_{t+m}(y)} dp_{t \otimes t+m}(x,y)$$

- Could use plug in estimator
- Tried k-nn type estimate of Kraskov, Stögbauer, and Grassberger 2004

$$\widehat{I(X,Y)} = \psi(k) - \frac{1}{n} \sum_{i=1}^{n} \left[\psi(n_x(i)+1) + \psi(n_y(i)+1) \right] + \psi(n)$$

where $\psi(\cdot)$ is the digamma function

Estimating information mixing—results



Inside the DSGE black box (linear)

1. Find the first order conditions for the system
2. Holding the stochastic process constant, derive the steady state of the system
3. Perform a (first order) Taylor series expansion around the steady state
4. Use the method of Sims 2001 to convert to linear state space form (basically a QZ decomposition)

Inside the DSGE black box (non-linear)

1. Find the first order conditions for the system
2. Holding the stochastic process constant, derive the steady state of the system
3. Use either of
   - Perturbation methods (higher order Taylor series expansions) using Schmitt-Grohé and Uribe 2004
   - Expand the offending nonlinear functions in some basis (projection methods) using Judd 1998

Blah blah

$$\begin{aligned} \mathbf{x}_t &= g(\boldsymbol{\alpha}_t, \epsilon_t) \\ \boldsymbol{\alpha}_{t+1} &= h(\boldsymbol{\alpha}_t, \eta_{t+1}) \\ \boldsymbol{\alpha}_1 &\sim F \end{aligned}$$