Empirical Finance: Methods & Applications

Economic Evaluation of Asset Returns

Pasquale Della Corte
Imperial College London

Week 3

Predictive Regressions

What is a Predictive Regression?

A predictive regression is a linear specification designed to assess the predictive power of past economic or financial variable for future asset returns.

A one-step ahead predictive regression is typically formulated as

$$\underbrace{y_t}_{\text{Asset return between times } t \text{ and } t-1} = \alpha + \underbrace{\beta}_{\text{Measures the significance of } x_{t-1} \text{ in predicting } y_t} \underbrace{x_{t-1}}_{\text{Lagged predictor observed at time } t} + \varepsilon_t$$

What's Wrong with a Predictive Regression?

Estimating a predictive regression is challenging in finance

- Predictors are highly persistent behaving like nearly nonstationary processes,
- Returns are generally noisier with fast mean-reverting dynamics,
- Shocks driving y_t and x_t often display a contemporaneous negative correlation.

The joint presence of persistence and contemporaneous correlation

- Distorts inferences based on standard normal (e.g., t-stat for β),
- Leads to spurious evidence of predictability.

Conventional in-sample analysis should be
complemented with an out-of-sample assessment.

What's Wrong with a Predictive Regression?

Consider a predictive regression with a persistent regressor

$$y_t = \alpha + \beta x_{t-1} + u_t$$

$$x_t = \mu + \rho x_{t-1} + v_t$$

When shocks to y_t and x_t are contemporaneously correlated

$$\mathbb{E}(u_t | x_t, x_{t-1}) \neq 0$$

This violation leads to bias in OLS estimates of β and challenges in finite-sample inference.

Finite-Sample Bias in OLS

Suppose shocks are normally distributed, independently across t, as

$$\begin{bmatrix} u_t \\ v_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_u^2 & \sigma_{uv} \\ \sigma_{uv} & \sigma_v^2 \end{bmatrix} \right)$$

Stambaugh (1999) derives the finite-sample distribution of $\hat{\beta} - \beta$ as

$$\mathbb{E}(\hat{\beta} - \beta) = \frac{\mathsf{Cov}(u_t, v_t)}{\mathsf{Var}(v_t)} \mathbb{E}(\hat{\rho} - \rho)$$

The assumption of normality is crucial for deriving the closed-form approximation of the bias.

Bias Formula (Stambaugh, 1999)

Stambaugh (1999) provides an approximation for the bias

$$\text{Bias}(\hat{\beta}) \approx -\frac{\text{Cov}(u_t, v_t)}{\text{Var}(v_t)} \times \frac{1+3\rho}{T}$$

What are the implications?

- The bias increases with higher persistence (as ρ approaches 1),
- The bias decreases with larger sample sizes (as $T \to \infty$).

How about the direction of the bias?

- β is upward biased when $\text{Cov}(u_t, v_t) < 0$,
- β is downward biased when $\text{Cov}(u_t, v_t) > 0$.

An Example: Negative Correlation

```
# Set the seed for reproducibility
Random.seed!(9876543)

# Parameters for the simulation
T          = 100      # Sample size
c          = 0.0      # Intercept
rho        = 0.99     # Persistence of x_t
a          = 0.0      # Intercept
beta       = 1.0      # True slope
sigma_u    = 3.0      # Std dev of u_t
sigma_v    = 1.0      # Std dev of v_t
sigma_uv   = -2.97    # Covariance between u_t and v_t
nsim       = 1000     # Number of Monte Carlo simulations

# Pre-allocate storage
bmat       = zeros(nsim)
bias       = zeros(nsim)

```

An Example: Negative Correlation

```
    # Covariance matrix and noise distribution
    S         = [sigma_u^2 sigma_uv; sigma_uv sigma_v^2]  # Covariance matrix
    L         = cholesky(S).L                              # Lower Cholesky

    # Start Monte Carlo simulation
    for sim in 1:nsim

      # Generate correlated errors
      u = zeros(T)
      v = zeros(T)

      for t in 1:T
        e     = L * randn(2)          # Apply Cholesky transformation
        u[t]  = e[1]                  # Extract correlated u_t
        v[t]  = e[2]                  # Extract correlated v_t
        end

```

An Example: Negative Correlation

```
# Generate y and x using AR(1) process
y = zeros(T)
x = zeros(T)

for t in 1:T
  if t == 1
    x0   = c / (1 - rho)
    x[t] = c + rho*x0   + v[t]
    y[t] = a + beta*x0 + u[t]
  else
    x[t] = c + rho *  x[t-1] + v[t]
    y[t] = a + beta * x[t-1] + u[t]
  end
end
```

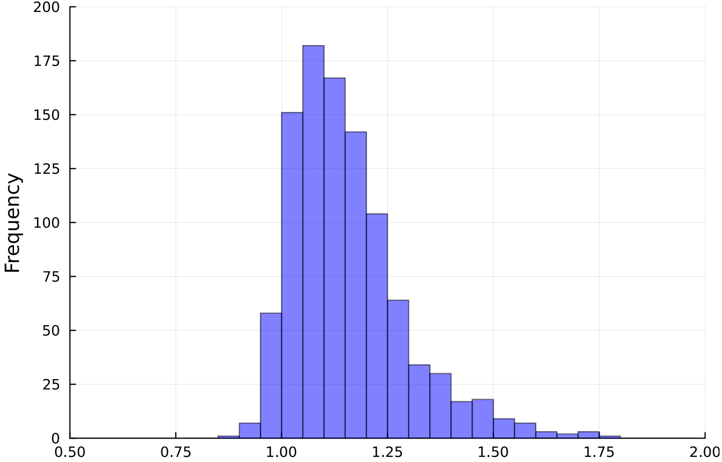
An Example: Negative Correlation

```
    # OLS regression
    y_reg      = y[2:end]
    x_reg      = [ones(T-1) x[1:end-1]]
    beta_ols   = inv(x_reg'*x_reg)*x_reg'*y_reg
    bmat[sim] = beta_ols[2]

    # Estimated bias
    bias[sim]  = beta_ols[2] - beta
  end

  # Compute results
  avg_beta = mean(bmat)
  avg_bias = mean(bias)

  # Approximated bias
  app_bias = - (sigma_uv / sigma_v^2) * ( (1 + 3 * rho) / T)
```

An Example: Negative Correlation

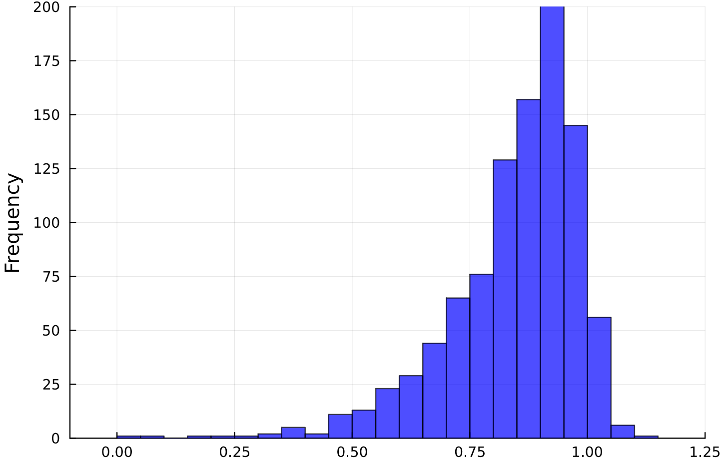
```

# Approximated bias
app_bias = - (sigma_uv / sigma_v^2) * ( (1 + 3 * rho) / T)

# Print results
println("Average OLS estimate: $avg_beta")
println("Average OLS  bias: $avg_bias")
println("Average Appr bias: $app_bias")


# Plot histogram
histogram(bmat, bins=30, color=:blue, alpha=0.7, label="", xlabel="",
ylabel="Frequency", title="")

```

OLS β with Negative Correlation



Simulations based on a Julia script: $Cov(u_t, v_t) = -2.97$, ave$(\beta) = 1.155$, and bias$(\beta) = 0.155$.

OLS β with Positive Correlation



Simulations based on a Julia script: $Cov(u_t, v_t) = 2.97$, ave$(\beta) = 0.846$, and bias$(\beta) = -0.154$.

Out-of-Sample Predictability

How to Assess Out-of-Sample Predictability?

Real-time information

- Only use information available to an investor when the forecast is produced,
- This is very hard with macro predictors due to revisions and release lags.

Horse race analysis

- Generate forecasts and assess performance in the next period,
- Compare the predictive ability of your model against a reference model,

Performance Evaluation

- Statistical or Economic Criteria to evaluate the performance of your models.

Which Benchmark?

A common benchmark is the random walk (nested with $\beta = 0$)

$$\underbrace{y_t}_{\text{Asset return from } t \text{ and } t-1} = \underbrace{\alpha}_{\text{Historical mean observed at time } t} + \varepsilon_t$$

Or the naïve random walk (nested with $\alpha = \beta = 0$)

$$\underbrace{y_t}_{\text{Asset return from } t \text{ and } t-1} = \underbrace{\varepsilon_t}_{\text{Completely unpredictable at time } t}$$

Our Competing Model

A one-step ahead predictive regression

$$y_t = \alpha + \beta\, x_{t-1} + \varepsilon_t$$

Asset return from t and $t-1$

Lagged predictor observed at time t

How much x_{t-1} helps predict y_t

The competing model is nesting the benchmark by setting

- $\beta = 0$ for the random walk, or
- $\alpha = \beta = 0$ for the naïve random walk.

Out-of-Sample Predictability

Suppose you have T observations (e.g., monthly from 1976/01 to 2024/12)

1. Estimate the model using a given training period (*in-sample*),
2. Generate a one-step ahead forecast of the next period return (*out-of-sample*),
3. Repeat these steps till the end of the sample.

We can generate out-of-sample forecasts using two methods

- Expanding window: in-sample data increase as you move forward, or
- Rolling window: in-sample data is fixed as you move forward,
- Suppose our initial training sample goes from 1976/01 to 1985/12.

Expanding Window: Step 1

OOS Predictability: Expanding Window

At time t (in Dec 1985): Run the following predictive regression

$$y_t = \alpha + \beta x_{t-1} + \varepsilon_t$$

returns up to t
(e.g. Feb 1976 to Dec 1985)

predictor up to t-1
(e.g. Jan 1976 to Nov 1985)

Estimate the parameters via OLS as

$$\widehat{\beta} = \frac{Cov(y_t, x_{t-1})}{Var(x_{t-1})} \quad \text{and} \quad \widehat{\alpha} = \overline{y} - \widehat{\beta}\,\overline{x}$$

sample mean
from 2 to t

sample mean
from 1 to t-1

OOS Predictability: Expanding Window

At time t (in Dec 1985): Construct the out-of-sample forecast as

or simply $\widehat{y}_{t+1}$ $\longleftarrow$ $$E_t\left(y_{t+1}\right) = \widehat{\alpha} + \widehat{\beta}\, x_t$$

forecast for t+1
(e.g. Jan 1986)

predictor in t
(e.g. Dec 1985)

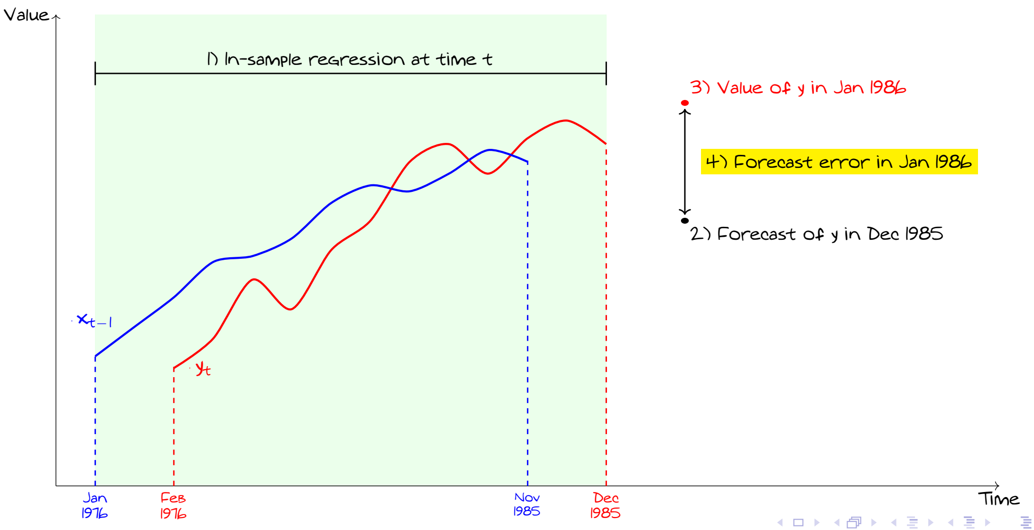
At time t+1 (in Jan 1986): Construct the forecast error as

$$\widehat{\varepsilon}_{t+1} = y_{t+1} - \widehat{y}_{t+1}$$

forecast errort at t+1
(e.g. Jan 1986)

actual return in t+1
(e.g. Jan 1986)

OOS Predictability: Expanding Window



Expanding Window: Step 2

OOS Predictability: Expanding Window

At time t+1 (in Jan 1986): Run the following predictive regression

$$\underbrace{y_{t+1}}_{\substack{\text{returns up to t+1} \\ \text{(e.g. Feb 1976 to Jan 1986)}}} = \alpha + \beta \underbrace{x_t}_{\substack{\text{predictor up to t} \\ \text{(e.g. Jan 1976 to Dec 1985)}}} + \varepsilon_{t+1}$$

Estimate the parameters via OLS as

$$\widehat{\beta} = \frac{Cov(y_{t+1}, x_t)}{Var(x_t)} \quad \text{and} \quad \widehat{\alpha} = \underbrace{\overline{y}}_{\substack{\text{sample mean} \\ \text{from 2 to t+1}}} - \widehat{\beta}\, \underbrace{\overline{x}}_{\substack{\text{sample mean} \\ \text{from 1 up to t}}}$$

OOS Predictability: Expanding Window

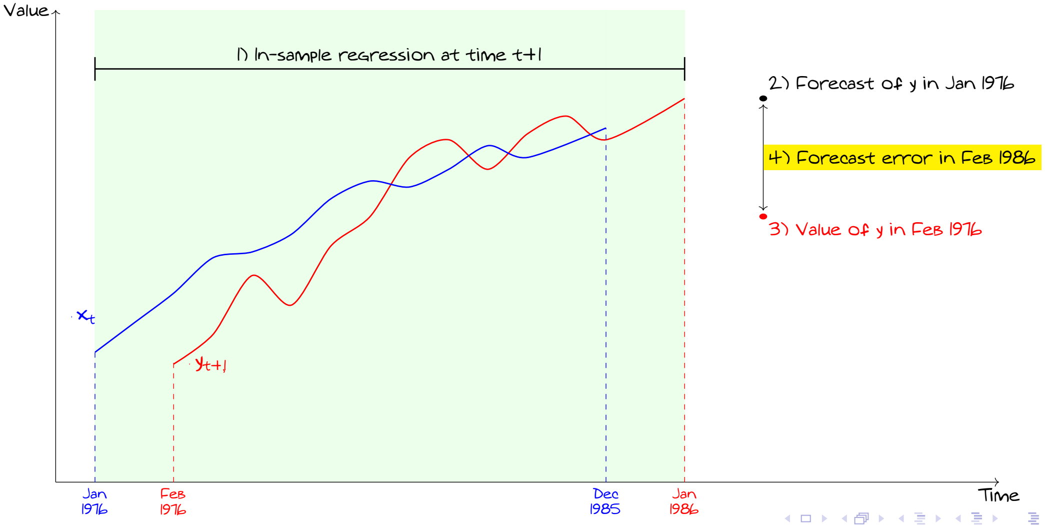
At time t+1 (in Jan 1986): Construct the out-of-sample forecast as

or simply $\widehat{y}_{t+2} \longleftarrow \underbrace{E_{t+1}\left(y_{t+2}\right)}_{\substack{\text{forecast for t+2} \\ \text{(e.g. Feb 1986)}}} = \widehat{\alpha} + \widehat{\beta}\underbrace{x_{t+1}}_{\substack{\text{predictor in t+1} \\ \text{(e.g. Jan 1986)}}}$

At time t+2 (in Feb 1986): Construct the forecast error as

$$\underbrace{\widehat{\varepsilon}_{t+2}}_{\substack{\text{forecast errort at t+2} \\ \text{(e.g. Feb 1986)}}} = \underbrace{y_{t+2}}_{\substack{\text{actual return in t+2} \\ \text{(e.g. Feb 1986)}}} - \widehat{y}_{t+2}$$

OOS Predictability: Expanding Window



... to the End

OOS Predictability: Expanding Window

```
# Control Variables
wind = 120                # Window size in months

# Load Data
df = CSV.read("data/data.csv", DataFrame)

# Extract Data
ydate = df.date
ydata = df.y
xdata = df.x

# Number of observations
T     = length(ydata)
```

OOS Predictability: Expanding Window

```
    # Predictive Regression Model
    mod_fmat = fill(-9999.0, T)                    # Forecast matri
    mod_emat = fill(-9999.0, T)                    # Error matrix

    for t in wind:T-1

    # Data up to time t
    y = ydata[2:t]
    x = xdata[1:t-1]

    # OLS estimates
    b = cov(x, y) / var(x)
    a = mean(y) - b * mean(x)

    # Save results (stored at t+1 for forecast made at time t)
    mod_fmat[t+1] = a + b * xdata[t]
    mod_emat[t+1] = ydata[t+1] - mod_fmat[t+1]

    end
```

OOS Predictability: Expanding Window

```
# Predictive Regression Benchmark
ben_fmat = fill(-9999.0, T)                 # Forecast matrix
ben_emat = fill(-9999.0, T)                 # Error matrix

for t in wind:T-1

# Data up to time t
y = ydata[2:t]

# OLS estimates
a = mean(y)

# Save results (stored at t+1 for forecast made at time t)
ben_fmat[t+1] = a
ben_emat[t+1] = ydata[t+1] - ben_fmat[t+1]

end
```

How About the Rolling Window?

OOS Predictability: Rolling Window

```
# Predictive Regression Model
mod_fmat = fill(-9999.0, T)                    # Forecast matrix
mod_emat = fill(-9999.0, T)                    # Error matrix
i        = 0

for t in wind:T-1

y = ydata[2+i:t]
x = xdata[1+i:t-1]

b = cov(x, y) / var(x)
a = mean(y) - b * mean(x)

mod_fmat[t+1] = a + b * xdata[t]
mod_emat[t+1] = ydata[t+1] - mod_fmat[t+1]
i             = i + 1

end
```

OOS Predictability: Rolling Window

```
# Predictive Regression Benchmark
ben_fmat = fill(-9999.0, T)             # Forecast matrix
ben_emat = fill(-9999.0, T)             # Error matrix
i        = 0

for t in wind:T-1

y = ydata[2+i:t]
a = mean(y)

ben_fmat[t+1] = a
ben_emat[t+1] = ydata[t+1] - ben_fmat[t+1]
i             = i + 1

end
```

Economic Restrictions

Campbell and Thompson (2008) recommend economic restrictions

- Often theory suggests the sign of the forecasts,
- Impose economically motivated sign restrictions.

Risk-based explanations predict a positive expected excess return when risk is high

1. Set $\widehat{\beta} = 0$ when $\widehat{\beta} < 0$,
2. Set $\widehat{y}_{t+1} = 0$ when $\widehat{y}_{t+1} < 0$,
3. Set $\widehat{\beta} = 0$ when $\widehat{\beta} < 0$ and $\widehat{y}_{t+1} = 0$ when $\widehat{y}_{t+1} < 0$.

OOS Predictability: Expanding Window + Sign Restrictions

```
# Control Variables
wind = 120                  # Window size in months
signr = 1                   # Sign restriction flag (1, 2, or 3

# Load Data
df = CSV.read("data/data.csv", DataFrame)

# Extract Data
ydate = df.date
ydata = df.y
xdata = df.x
T     = length(ydata)
```

OOS Predictability: Expanding Window + Sign Restrictions

```
# Predictive Regression Model
mod_fmat = fill(-9999.0, T)                # Forecast matrix
mod_emat = fill(-9999.0, T)                # Error matrix

for t in wind:T-1

    # Data up to time t
    y = ydata[2:t]
    x = xdata[1:t-1]

    # OLS estimates
    b = cov(x, y) / var(x)
    a = mean(y) - b * mean(x)




```

OOS Predictability: Expanding Window + Sign Restrictions

```
    # Sign restrictions (assuming un thoery b > 0 and yhat > 0)
    if signr == 1                   # set b = 0 if b < 0
    b     = max(b, 0.0)
    yhat = a + b * xdata[t]

    elseif signr == 2           # set yhat = 0 if yhat < 0
    yhat = max(a + b * xdata[t], 0.0)

    elseif signr == 3          # set b = 0 if b < 0 and yhat = 0 if yhat < 0
    b     = max(b, 0.0)
    yhat = max(a + b * xdata[t], 0.0)
    end

    # Save results (stored at t+1 for forecast made at time t)
    mod_fmat[t+1] = yhat
    mod_emat[t+1] = ydata[t+1] - mod_fmat[t+1]

    end
```

Statistical Evaluation

Statistical Evaluation: MSE

<u>Mean Squared Error (MSE)</u>: Minimize the average squared forecast errors

$$MSE = \frac{1}{T-t}\sum_{i=t+1}^{T} \hat{\varepsilon}_i^2$$

Computed for the model (MOD)
and the benchmark (BEN)

Average of squared
forecast errors

A lower MSE produces better out-of-sample forecasts:
Pick the model when $MSE_{MOD} < MSE_{BEN}$.

Statistical Evaluation: MAE

Mean Absolute Error (MAE): Minimize the average absolute forecast errors

$$\boxed{MAE} = \boxed{\frac{1}{T-t}\sum_{i=t+1}^{T} |\widehat{\varepsilon}_i|}$$

Computed for the model (MOD)
and the benchmark (BEN)

Average of absolute
forecast errors

A lower MAE produces better out-of-sample forecasts:
Pick the model when $MAE_{MOD} < MAE_{BEN}$.

Statistical Evaluation: R^2_{OOS}

Out-of-sample R-squared (R^2_{OOS}): Suggested by Campbell & Thompson (2008)

MSE for the model

$$R^2_{oos} = 1 - \frac{MSE_{MOD}}{MSE_{BEN}}$$

Out-of-sample R^2

MSE for the benchmark

$R^2_{oos} > 0$ means that MOD outperforms BEN with a lower MSE.

Statistical Evaluation: Clark & West Test

How about testing the null of equal predictive ability?

$$H_0 : MSE_{BEN} = MSE_{MOD}$$

Clark and West (2007) note that this hypothesis is incorrectly formulated

Fewer parameters to estimate

Additional parameters, possibly useless for prediction, add estimation error

If MOD and BEN were equally good, $MSE_{MOD} > MSE_{BEN}$

Statistical Evaluation: Clark & West Test

Clark and West (2007) propose a test designed for nested models

$$\underbrace{f_{t+1}}_{\text{Statistic}} = \underbrace{\widehat{\varepsilon}^2_{t+1,BEN}}_{\text{Forecast error Benchmark}} - [\underbrace{\widehat{\varepsilon}^2_{t+1,MOD}}_{\text{Forecast error model}} - \underbrace{(\widehat{y}_{t+1,BEN} - \widehat{y}_{t+1,MOD})^2}_{\text{Correction Term}}]$$

Construct this statistic for $f_{t+1}, f_{t+2}, \ldots f_T$

Statistical Evaluation: Clark & West Test

Regress f on a constant

$$\underbrace{f_i}_{\substack{\text{Statistic for}\\ i=t+1,t+2,\dots,T}} = \underbrace{\rho}_{\text{Constant Term}} + \eta_i$$

Construct the t-test on ρ

$$CW = \frac{\widehat{\rho}}{se(\widehat{\rho})}$$

The finite-sample distribution is non-standard but converges to a Normal in large samples (e.g., T>100)

If CW >1.65, reject H_0 at the 5% level (one-sided test)

A bootstrap procedure is common when the sample is small.

OOS Predictability: Statistical Evaluation

```
#Construct MSE
ind      = findall(mod_fmat .!= -9999)
MSE_mod  = mean(mod_emat[ind].^2)
MSE_ben  = mean(ben_emat[ind].^2)

# Construct R2_OOS
R2_oos = (1 .- (MSE_mod ./ MSE_ben))

# Construct CW
ben_sqre = (ben_emat[ind].^2)
mod_sqre = ( (mod_emat[ind].^2) .- (ben_fmat[ind] - mod_fmat[ind]).^2)
fstat    = ben_sqre - mod_sqre
T        = length(fstat)
```

OOS Predictability: Statistical Evaluation

```julia
# OLS regression on a constant
df_cw = DataFrame(fstat = fstat)
model = lm(@formula(fstat ~ 1), df_cw)

# Calculate Newey-West Covariance
lag    = floor(Int, 4 * (T/100)^(2/9))
nw_cov = vcov(Bartlett(lag), model)
t_stat = coef(model)[1] / sqrt(nw_cov[1,1])

# Output Results
println("MSE (Model): ", MSE_mod)
println("MSE (Benchmark): ", MSE_ben)
println("R2_OOS: ", r2_oos)
println("CW: ", t_stat)


```

Market Timing: Henriksson & Merton Test

Market Timing Test: Proposed by Henriksson & Merton (1981)

number of correct forecasts
(same sign as actual returns)

$$p = \frac{c}{n}$$

probability of
directional accuracy

total number of forecasts

Hypothesis Testing

$$H_0 : p = 0.5 \quad \text{(no market timing)}$$

$$H_A : p > 0.5 \quad \text{(positive market timing)}$$

Market Timing: Henriksson & Merton Test

p is distributed as a Binomial distribution with

$$E[p] = c/n$$

$$Var[p] = \frac{p(1-p)}{n}$$

For large samples, the Binomial is approximately Normal and

$$Z = \frac{E[p] - 0.5}{\sqrt{Var[p]}} \sim N(0,1)$$

If $Z > 1.65$, reject H_0 at the 5% level (one-sided test)

Bootstrap Algorithm

Bootstrap Algorithm

Following Mark (1995) and Kilian (1999), the algorithm proceeds as follows:

1. Using actual data, compute the test statistic of interest $\widehat{\tau}$ (e.g., R^2_{OOS}, CW, etc).

2. Using OLS, estimate the following model under the null of no predictability:

$$
\begin{aligned}
y_t &= \alpha + u_{1,t} \\
x_t &= c + \rho_1 x_{t-1} + \ldots + \rho_p x_{t-p} + u_{2,t},
\end{aligned}
$$

with parameters based on the full sample and p selected by BIC.

Bootstrap Algorithm

3. Simulate data as follows:

$$y_t^\star = \widehat{\alpha} + u_{1,t}^\star$$
$$x_t^\star = \widehat{c} + \widehat{\rho}_1 x_{t-1}^\star + \ldots + \widehat{\rho}_p x_{t-p}^\star + u_{2,t}^\star,$$

   where $u_t^\star = (u_{1,t}^\star, u_{2,t}^\star)'$ are drawn with replacement from the estimated residuals $\widehat{u}_t = (\widehat{u}_{1,t}, \widehat{u}_{2,t})'$, and $(x_{t-1}^\star, \ldots, x_{t-p}^\star)'$ is a block of consecutive data randomly drawn from the actual data. Each simulated sample has the same size as the original sample.

4. Using simulated data, compute $\widehat{\tau}^\star$ from OOS forecasts of BEN and MOD. Use the same procedure as for the actual data.

Bootstrap Algorithm

5. Repeat steps 3 and 4 for $B = 1{,}000$ (or more).

6. Compute the one-sided p-value of $\widehat{\tau}$ as:

$$p\text{-value} = \frac{1}{B} \sum_{j=1}^{B} I(\widehat{\tau}_j^{\star} > \widehat{\tau}),$$

   where $I(\cdot)$ is equal to 1 when its argument is true and 0 otherwise.

Economic Evaluation

Economic Evaluation

"Economists are often puzzled as to why profit-maximizing firms buy economic forecasts. Summary statistics [. . .] **rarely reveal major differences** *between professional forecasting services and a simple naive approach of no change in the variable being forecast... Yet, millions of dollars are spent annually both producing and purchasing these apparently worthless forecasts."*

Leitch and Tanner, *American Economic Review*, 1992

"The standard criteria show no consistent relationship with profits. Indeed, MAE and RMSE criteria have perverse signs (better forecasts should have lower average errors and higher profits, and thus, the simple correlations should be negative)."

Leitch and Tanner, *American Economic Review*, 1992

Economic Evaluation

A statistical analysis does not necessarily reveal the economic value of an active strategy.

- Consider an investor allocating wealth between a riskless asset and N risky assets.

To simplify the notation, let's define

$$\begin{bmatrix} \widehat{y}_{1,t+1} \\ \widehat{y}_{2,t+1} \\ \vdots \\ \widehat{y}_{N,t+1} \end{bmatrix} \qquad \begin{bmatrix} w_{1,t} \\ w_{2,t} \\ \vdots \\ w_{N,t} \end{bmatrix} \qquad \begin{bmatrix} \widehat{\sigma}^2_{1,t+1} & \widehat{\sigma}_{12,t} & \cdots & \widehat{\sigma}_{1N,t} \\ \widehat{\sigma}_{21,t} & \widehat{\sigma}^2_{2,t} & \cdots & \widehat{\sigma}_{2N,t} \\ \vdots & \cdots & \ddots & \cdots \\ \widehat{\sigma}_{N1,t} & \widehat{\sigma}_{N2,t} & \cdots & \widehat{\sigma}^2_{N,t} \end{bmatrix}$$

vector of return forecasts at time t
$\widehat{\boldsymbol{r}}_{t+1}$

vector of weights to calculate at time t
$\boldsymbol{w}_t$

matrix of variance-covariance forecasts at time t
$\widehat{\boldsymbol{\Sigma}}_{t+1}$

Economic Evaluation

Vector of return forecasts at time t, or simply $\hat{\boldsymbol{r}}_{t+1}$

- For each risky asset, run the predictive regression at time t, and generate the out-of-sample forecast for time $t+1$.

Matrix of variance-covariance forecasts at time t, or simply $\widehat{\Sigma}_{t+1}$

- For each risky asset, get the regression residuals at time t, calculate the sample variance-covariance matrix, and use it for time $t+1$ (alternatively, you can use EWMA, DCC, etc).

Vector of weights to calculate at time t, or simply $\boldsymbol{w}_t$

- We will employ the maximum return portfolio strategy based on mean variance.

Maximum Expected Return

Each period t, the investor solves the following problem

vector of ones

portfolio expected return

$$\max_{\boldsymbol{w}_t} \quad \widehat{r}_{p,t+1} = \boldsymbol{w}_t' \widehat{\boldsymbol{r}}_{t+1} + (1 - \boldsymbol{w}_t' \boldsymbol{\iota})\, r_{f,t}$$

riskless return

$$\text{subject to} \quad \sigma_p^\star = \text{target volatility}$$

for example, 10% per annum

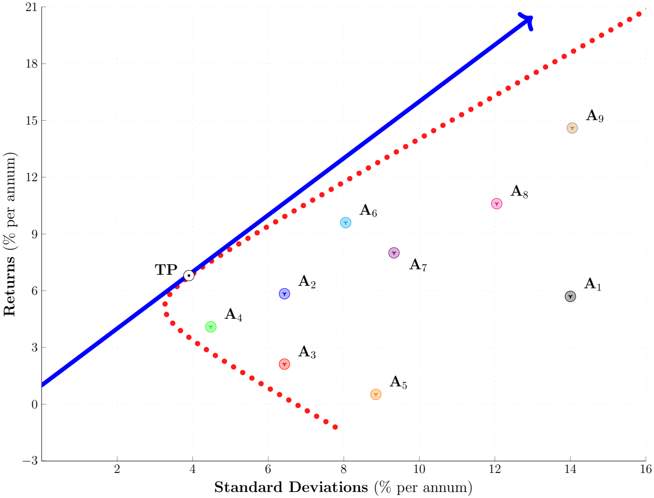
The solution for $\boldsymbol{w}_t$ is in closed-form

$$\boldsymbol{w}_t = \frac{\sigma_p^*}{\sqrt{C_t}} \widehat{\boldsymbol{\Sigma}}_{t+1}^{-1} (\widehat{\boldsymbol{r}}_{t+1} - r_{f,t})$$

where

$$C_t = (\widehat{\boldsymbol{r}}_{t+1} - r_{f,t})' \widehat{\boldsymbol{\Sigma}}_{t+1}^{-1} (\widehat{\boldsymbol{r}}_{t+1} - r_{f,t}).$$

Maximum Return Strategy: An Illustration



Maximum Return Strategy

At time $t+1$, compute the realized portfolio return as

$$r_{p,t+1} = \boldsymbol{w_t}' \boldsymbol{r}_{t+1} + \left(1 - \boldsymbol{w_t}' \boldsymbol{\iota}\right) r_{f,\,t}$$

realized portfolio return
at time t+1

vector of realized
risky returns at time t+1

Continue until the end of the sample

Performance Evaluation

The Sharpe ratio (SR) quantifies the excess return per unit of volatility

Measures meanvariance trade-off but ignores higher moment risks $\longleftarrow$

$$SR = \frac{E\left(r_{p,t+1} - r_{f,\,t}\right)}{\sqrt{Var\left(r_{p,t+1} - r_{f,\,t}\right)}}$$

average portfolio excess returns

volatility portfolio excess returns

The Sortino ratio (SO) quantifies the excess return per unit of bad volatility

Differentiates between 'ups and downs' in returns. A large SO implies low risk of large losses. $\longleftarrow$

$$SO = \frac{E\left(r_{p,t+1} - r_{f,\,t}\right)}{\sqrt{Var\left(r_{p,t+1} - r_{f,\,t} < 0\right)}}$$

average portfolio excess returns

downside volatility portfolio excess returns

Performance Evaluation

The certainty equivalent return of Goetzmann, Ingersoll, Spiegel & Welch (2007)

Degree of relative risk aversion

What makes you indifferent between a sure return and a risky investment strategy.

$$\longleftarrow CER = \frac{1}{1-\delta} \ln \left\{ E \left[\left(\frac{1 + r_{p,t+1}}{1 + r_{f,\,t}} \right)^{1-\delta} \right] \right\}$$

Consider the difference of performance measures as

$$\mathcal{P} = CER_{MOD} - CER_{BEN}$$

The performace fee an investor would pay to replace BEN with MOD

Transaction Costs Impact

The break-even transaction cost makes an investor indifferent between two models

$$\tau_{t+1} = \frac{1}{N}\sum_{i=1}^{N}\left| w_{i,t+1} - \boxed{w_{i,t}\left(\frac{1+r_{i,t+1}}{1+r_{p,t+1}}\right)}\right|$$

Weight on asset i at time t+1 after rebalancing

Weight on asset i at time t+1 before rebalancing

Weight on asset i at time t after rebalancing

and then

$$\tau^{be} = \frac{E(r_{p,t+1}^{MOD} - r_{p,t+1}^{BEN})}{E(\tau_{p,t+1}^{MOD} - \tau_{p,t+1}^{BEN})}$$

An investor prefers MOD to BEN if she pays less than τ^{be}.

Example: A Portfolio with Two Assets

At time t, mean–variance optimization implies an equal allocation

$$w_{A,t} = 0.50 \qquad \text{and} \qquad w_{B,t} = 0.50$$

Between times t and $t+1$, A increases by 20% and B remains unchanged

$$w'_{A,t+1} = 0.55 \qquad \text{and} \qquad w'_{B,t+1} = 0.45$$

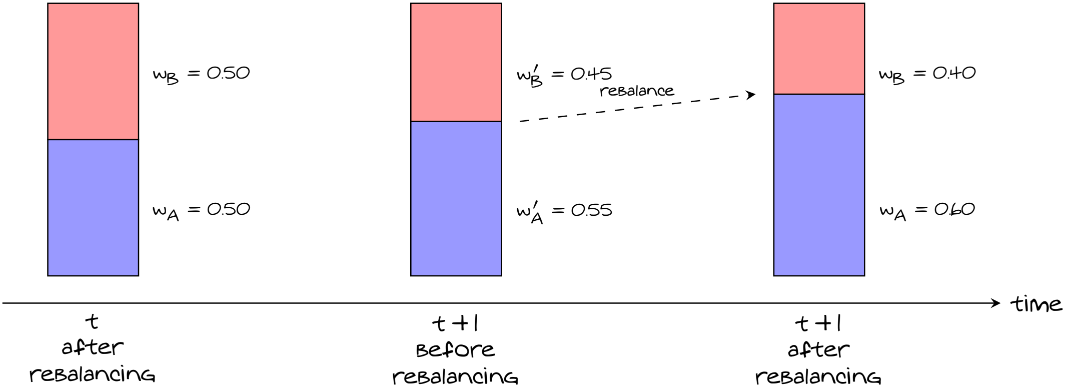
At time $t+1$, mean–variance optimization prescribes target weights:

$$w_{A,t+1} = 0.60 \qquad \text{and} \qquad w_{B,t+1} = 0.40$$

Transaction costs arise from the required rebalancing trades:

$$\tau_{t+1} = \frac{1}{2}\left(\left|0.60 - 0.55\right| + \left|0.40 - 0.45\right|\right)$$

Example: A Portfolio with Two Assets



Model Combination

Combined Forecasts

So far, we have evaluated the performance of individual models relative to the benchmark but ex-ante we ignore which model is true and this generates model uncertainty.

We account for model uncertainty using combined forecasts from the full set of predictive regressions. The superior performance of combined forecasts is known since the seminal work of Bates and Granger (1969).

We review three types of combined forecasts

- Simple model averaging,
- Statistical model averaging,
- Economic model averaging.

Combined Forecasts

For each risky asset

- We estimate predictive regressions for $n = 1, \ldots, M$ models,
- Each regression provides an individual forecast $\widehat{y}_{t+1}^{m}$.

The combined forecast $\widetilde{y}_{t+1}$ is a weighted average of individual forecasts

$$\widetilde{y}_{t+1} = \sum_{m=1}^{M} \kappa_t^m \times \widehat{y}_{t+1}^{m}$$

- κ_t^m is the ex-ante weight on m at time t,
- How do we compute these ex-ante model weights?

Simple Model Averaging

The **mean rule**

- For each forecast, set $\kappa_t^m = 1/M$.

The **median rule**

- Select the median across κ_t^m.

Statistical Model Averaging

At time t, construct the combining weights as

$$\kappa_t^m = \frac{1/\,MSE_t^m}{\sum_{m=1}^{M} 1/\,MSE_t^m}$$

- $\kappa_t^m \rightarrow$ a low MSE_t^m implies a high weight for the forecast n,
- $MSE_t^m \rightarrow$ mean-squared error based on data up to time t,
- See Bates and Granger (1969) and Stock and Watson (2004).

Economic Model Averaging

At time t, construct the combining weights as

$$\kappa_t^m = \frac{SR_t^m}{\sum_{n=1}^{M} SR_t^m}$$

- $\kappa_t^m \rightarrow$ a high SR_t^m implies a high weight for the forecast n,
- $SR_t^m \rightarrow$ Sharpe ratio based on data up to time t,
- See Della Corte and Tsiakas (2012).

A Simple Application

A Currency Fund Manager

Which model should an asset manager use for asset allocation?

- We answer this question by considering a set of models,
- We run out-of-sample tests using statistical and economic criteria.

Our benchmark model

- Random Walk with drift (RW)

Our competing model

- Purchasing Power Parity (PPP),
- Uncovered Interest Parity (UIP),
- Monetary Fundamentals (MF),
- Asymmetric Taylor rule (TR),

Predictive Regression

Consider the following one-step ahead predictive regression

$$y_t = \alpha + \beta x_{t-1} + u_t$$

where $y_t = s_t - s_{t-1}$ is the nominal exchange rate return.

Random Walk with drift (RW)

$$x_{t-1} = 0$$

Uncovered Interest Parity (UIP)

$$x_{t-1} = i_{t-1} - i^{\star}_{t-1}$$

Predictive Regression

Purchasing Power Parity (PPP)

$$x_{t-1} = p_{t-1} - p^{\star}_{t-1} - s_{t-1}.$$

Monetary Fundamentals (MF)

$$x_{t-1} = \left(m_{t-1} - m^{\star}_{t-1}\right) - \left(zt - 1 - z^{\star}_{t-1}\right) - s_{t-1}$$

Asymmetric Taylor Rule

$$x_{t-1} = 0.1 q_{t-1} + 0.1 \left(g_{t-1} - g^{\star}_{t-1}\right) + 1.5 \left(\pi_{t-1} - \pi^{\star}_{t-1}\right)$$

Data Sample

The sample data consists of

- End-of-month data ranging from January 1976 to June 2010,
- 9 currencies relative to the US dollar (USD): Australian dollar (AUD), Canadian dollar (CAD), Swiss franc (CHF), Deutsche mark-euro (EUR), British pound (GBP), Japanese yen (JPY), Norwegian kroner (NOK), New Zealand dollar (NZD), and Swedish kronor (SEK),

Our OOS period runs from January 1986 to December 2010

- First OOS forecast uses data from Jan-76 to Dec-85,
- Moving forward using an expanding window until May 2010.

Statistical Evaluation

	UIP	*PPP*	*MF*	*TR*	*UIP*	*PPP*	*MF*	*TR*
			AUD				**CAD**	
$R^2_{oos}(\%)$	-0.26	0.34	-1.21	0.91	0.06	0.30	-1.32	0.09
			CHF				**EUR**	
R^2_{oos} (%)	-1.68	1.11	-0.35	-2.37	-1.55	0.74	-0.73	-1.71
			GBP				**JPY**	
R^2_{oos} (%)	-1.22	1.22	-0.33	1.64	0.71	0.30	-1.67	-1.64
			NOK				**NZD**	
R^2_{oos} (%)	-0.50	0.68	-1.42	1.01	0.11	-0.37	-0.65	1.15
			SEK					
R^2_{oos} (%)	-1.00	0.72	-0.28	0.77				

Sources: Della Corte & Tsiakas (2012).

Economic Evaluation

	μ_p	σ_p	SR	SO	$\mathcal{P}$	τ^{be}
RW	10.8	11.4	0.54	0.73		
UIP	11.9	11.1	0.65	0.99	143	173
PPP	13.3	11.3	0.76	0.97	247	70
MF	11.1	11.7	0.55	0.73	4	-
TR	12.1	11.4	0.65	0.83	121	161

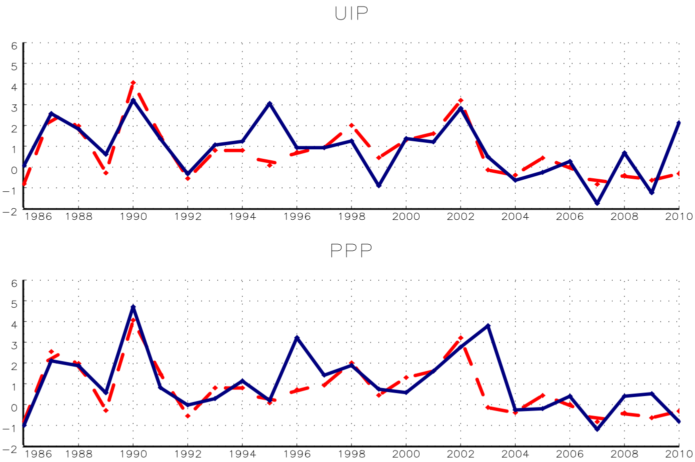
Sources: Della Corte & Tsiakas (2012).

Model Averaging

	μ	σ	SR	SO	$\mathcal{P}$	τ^{be}
Benchmark						
RW	10.8	11.4	0.54	0.73		
Simple Model Averaging						
Mean	13.3	11.6	0.74	0.89	206	81
Median	12.0	12.0	0.61	0.74	52	21
Statistical Model Averaging						
MSE	13.3	11.6	0.74	0.89	204	81
Economic Model Averaging						
SR	12.9	11.4	0.72	0.89	187	103

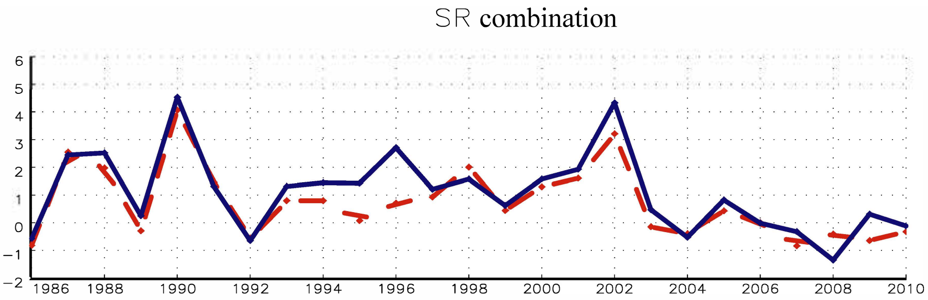
Sources: Della Corte & Tsiakas (2012).

Model Averaging



Source: Della Corte & Tsiakas (2012).

Model Averaging



Source: Della Corte & Tsiakas (2012).

The RW strategy in red and the competing strategy in blue