Financial Statistics in Stata

David Schenck

Stata

November 6, 2026
2026 Canadian Stata Conference

Outline

1. Financial data
2. Portfolios
3. Financial regressions
4. Value at Risk

The `fin` suite

- `finreturns`
- `finportfolio`
- `finsummarize`
- `finregress`
- `finvalrisk`

Outline

1. Financial data
2. Portfolios
3. Financial regressions
4. Value at Risk

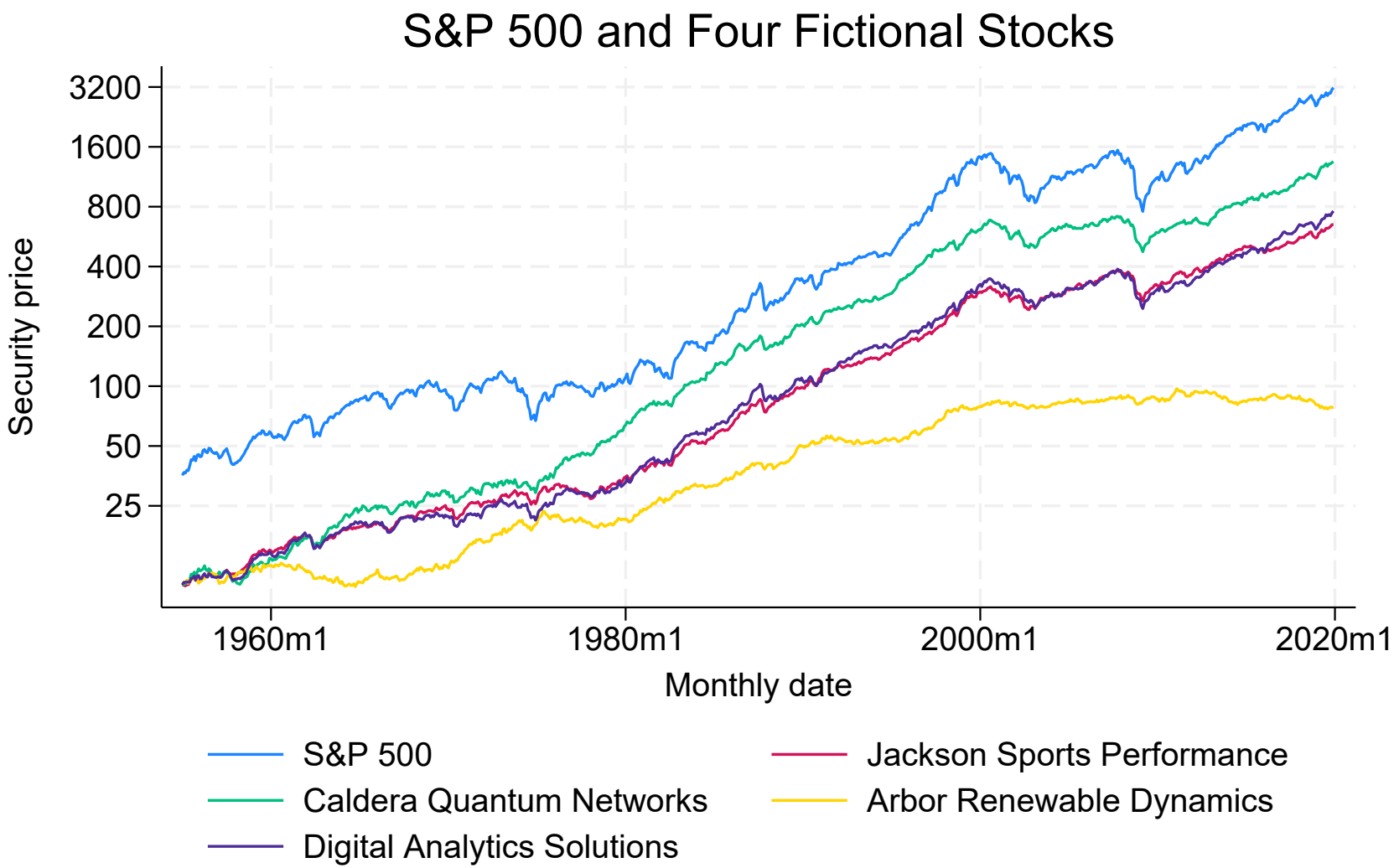
Financial data

```
. use finweb
(fictional stock price data)

. describe datem-tyr
```

Variable name	Storage type	Display format	Value label	Variable label
datem	int	%tm		Monthly date
sp500	double	%10.0g		S&P 500
vol	float	%9.0g		Volatility index
rf	float	%9.0g		Risk-free rate, monthly
jsp	float	%9.0g		Jackson Sports Performance
cqn	float	%9.0g		Caldera Quantum Networks
ard	float	%9.0g		Arbor Renewable Dynamics
das	float	%9.0g		Digital Analytics Solutions
acme	float	%9.0g		Aciron Medical, Inc.
bat	float	%9.0g		Boron Advanced Technologies
dune	float	%9.0g		Digital Urban Network Enterprise
tyr	float	%9.0g		Tyndale Resources Group

Financial data



From prices to returns

- New command `finreturns` generates returns from prices
- First difference, $P_t - P_{t-1}$
- Simple returns, $P_t/P_{t-1} - 1$
- Log returns $\ln(P_t/P_{t-1})$
- Annualized returns, $(P_t/P_{t-1})^m - 1$
- Log-annualized returns, $m \times \ln(P_t/P_{t-1})$

($m = 12$ for monthly data, $m = 4$ for quarterly data)

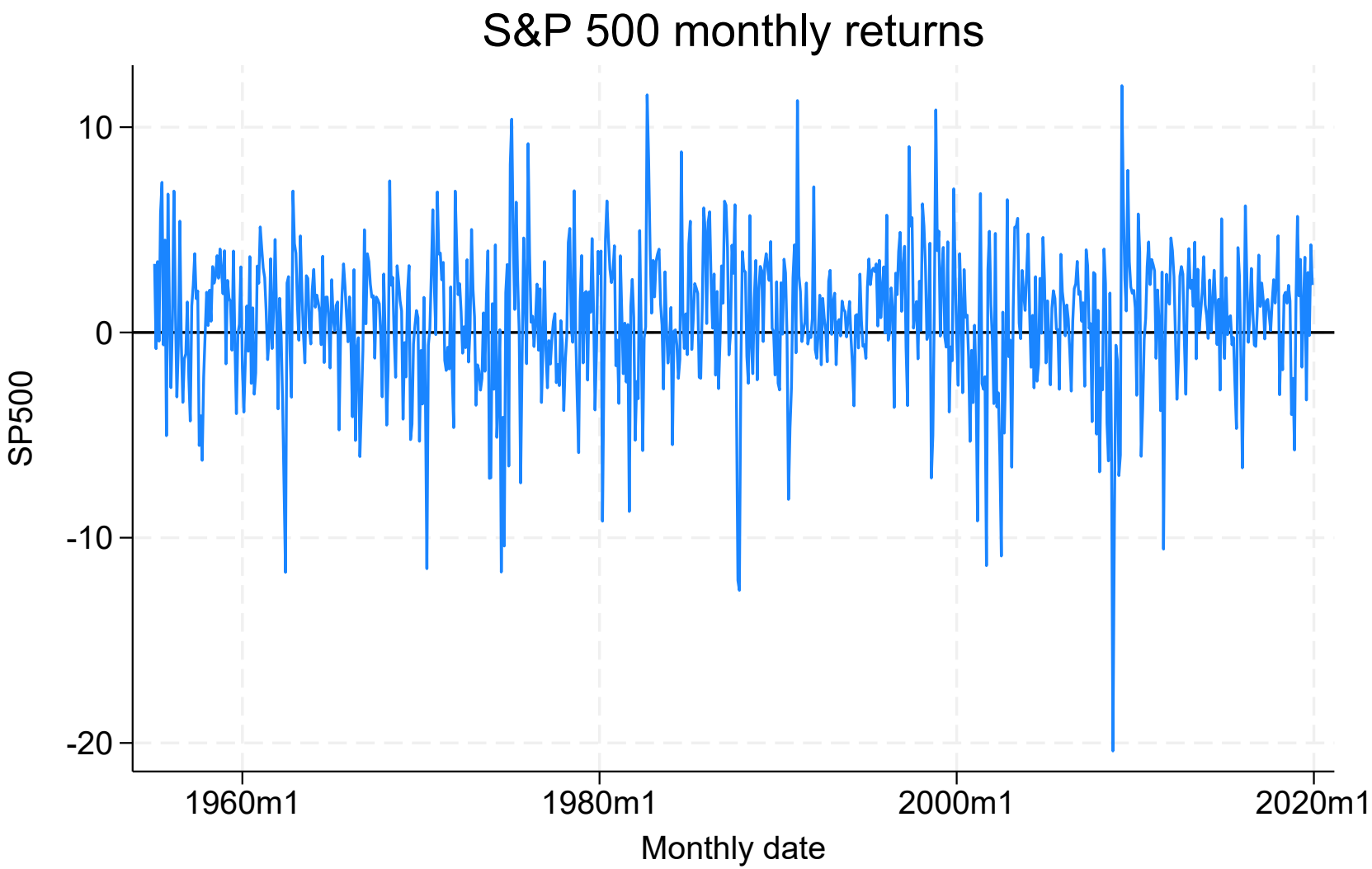
From prices to returns

```
. finreturns sp500, simple(r_mkt) multiply(100)
(all returns multiplied by 100)

Simple returns generated for variable sp500:
```

	datem	sp500	r_mkt
1.	1955m1	35.6	.
2.	1955m2	36.79	3.3426966
3.	1955m3	36.5	-.78825768
4.	1955m4	37.76	3.4520548
5.	1955m5	37.6	-.42372881
6.	1955m6	39.78	5.7978723
7.	1955m7	42.69	7.3152338
8.	1955m8	42.43	-.60904193
9.	1955m9	44.34	4.5015319
10.	1955m10	42.11	-5.0293189

Returns over time



From prices to returns

```
. finreturns jsp cqn ard das, multiply(100) simple(r_) nopreview
(all returns multiplied by 100)

Simple returns generated for variables jsp, cqn, ard, and das.

. describe r_mkt r_jsp r_cqn r_ard r_das
```

Variable name	Storage type	Display format	Value label	Variable label
r_mkt	double	%10.0g		SP500
r_jsp	double	%10.0g		JSP
r_cqn	double	%10.0g		CQN
r_ard	double	%10.0g		ARD
r_das	double	%10.0g		DAS

Outline

1 Financial data

2 Portfolios

3 Financial regressions

4 Value at Risk

Asset Returns

```
. summarize r_mkt r_jsp r_cqn r_ard r_das
```

Variable	Obs	Mean	Std. dev.	Min	Max
r_mkt	779	.6387889	3.458349	-20.39114	12.02171
r_jsp	779	.5583062	2.04691	-10.97514	7.364605
r_cqn	779	.6601578	2.400168	-10.59864	8.662755
r_ard	779	.282029	1.871475	-5.362448	6.507083
r_das	779	.5852287	2.350985	-13.07848	7.054804

From Assets to Portfolios

- A portfolio is a linear combination of assets
- Portfolio returns:

$$R_p = w_1 r_1 + w_2 r_2 + w_3 r_3$$

- w_i is the weight on asset i
- r_i is the return on asset i
- R_p is the return on the portfolio
- How to choose weights?
  - Equal weights
  - Weights chosen by the researcher
  - Minimum variance weights
  - Maximum return-to-risk (Sharpe ratio) weights

Constructing portfolios

- New command `finportfolio` creates portfolios
  - Equal weights with `finportfolio equal`

```
. finportfolio equal r_jsp r_cqn r_ard r_das,          ///
>         generate(p1, label("equal weight"))
Equally weighted portfolio
Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Weight
JSP	.25
CQN	.25
ARD	.25
DAS	.25

```
Portfolio return    = 0.5214
Portfolio std. dev. = 1.6854
Risk-free rate      = 0.0000
Sharpe ratio        = 0.3094
```

Constructing portfolios

- New command `finportfolio` creates portfolios
  - Supplied weights with `finportfolio fixed`

```
. finportfolio fixed  r_jsp r_cqn r_ard r_das, at(.5 .3 .2 0) ///
>         generate(p2, label("fixed"))
Fixed weights portfolio
Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Weight
JSP	.5
CQN	.3
ARD	.2
DAS	0

```
Portfolio return    = 0.5336
Portfolio std. dev. = 1.7007
Risk-free rate      = 0.0000
Sharpe ratio        = 0.3138
```

Constructing portfolios

- New command `finportfolio` creates portfolios
  - Minimum-variance weights with `finportfolio minvariance`

```
. finportfolio minvariance r_jsp r_cqn r_ard r_das, target(0.6) ///
>         generate(p3, label("minvar"))

Minimum variance portfolio with fixed return
Short selling allowed

Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Weight
JSP	.4080469
CQN	.6026235
ARD	.0639746
DAS	-.0746451

```
Portfolio return    = 0.6000 (fixed return)
Portfolio std. dev. = 1.9504
Risk-free rate      = 0.0000
Sharpe ratio        = 0.3076
```

Constructing portfolios

- New command `finportfolio` creates portfolios
  - Positive weights with `finportfolio minvariance, noshort`

```
. finportfolio minvar r_jsp r_cqn r_ard r_das, target(0.6) noshort nolog
Minimum variance portfolio with fixed return
Short selling not allowed

Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Weight
JSP	.3690342
CQN	.5710817
ARD	.0596439
DAS	.0002402

```
Portfolio return    = 0.6000 (fixed return)
Portfolio std. dev. = 1.9528
Risk-free rate      = 0.0000
Sharpe ratio        = 0.3073
```

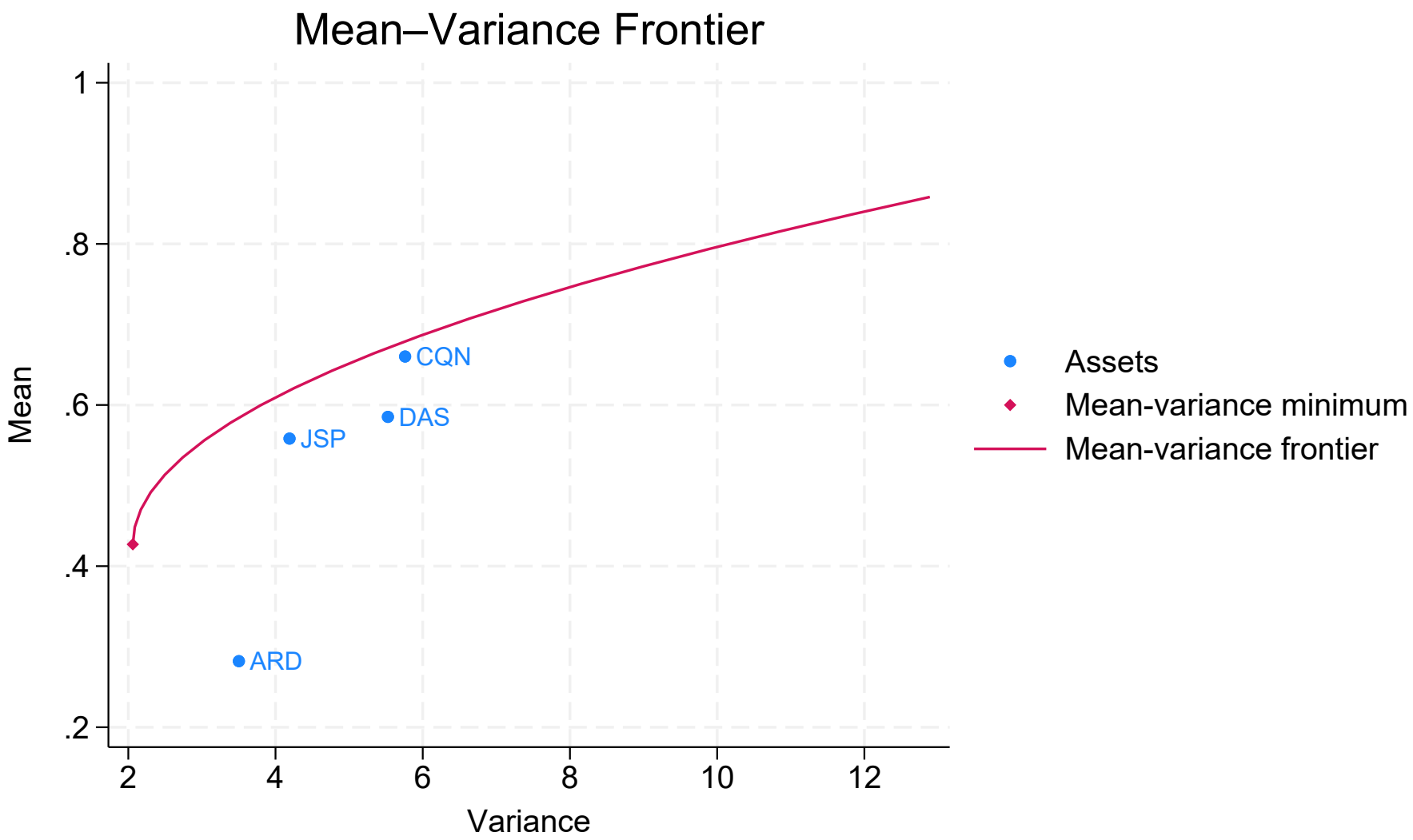
Comparing portfolios

- New command `finsummarize` compares portfolios

```
. finsummarize r_jsp r_cqn r_ard r_das p1 p2 p3
Financial summary statistics
Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Mean	Std. dev.	Sharpe ratio
JSP	0.5583	2.0469	0.2728
CQN	0.6602	2.4002	0.2750
ARD	0.2820	1.8715	0.1507
DAS	0.5852	2.3510	0.2489
equal weight	0.5214	1.6854	0.3094
fixed	0.5336	1.7007	0.3138
minvar	0.6000	1.9504	0.3076

The efficient frontier



Outline

1 Financial data

2 Portfolios

3 Financial regressions

4 Value at Risk

The Capital Asset Pricing Model

- We often wish to compare an asset's returns to a benchmark

$$r_{it} - r_t^f = \alpha_i + \beta_i(r_t^m - r_t^f) + u_t$$

- r_t^f is the risk-free rate
- r_t^m is the benchmark return
- α_i is the intercept for asset i
- β_i is the slope coefficient on asset i

finregress capm data preparation

```
. finreturns jsp-iron, log(lnr_) multiply(100) nopreview
(all returns multiplied by 100)

Log returns generated for variables jsp, cqn, ard, das, acme, bat, dune,
tyr, glo, spa, wgt, bar, yum, cph, dil, ege, epg, jml, khc, krg, kth, nhb,
tks, and iron.

. finreturns sp500, log(lnr_mkt) multiply(100)
(all returns multiplied by 100)

Log returns generated for variable sp500:
```

	datem	sp500	lnr_mkt
1.	1955m1	35.6	.
2.	1955m2	36.79	3.2880431
3.	1955m3	36.5	-.79138085
4.	1955m4	37.76	3.3938081
5.	1955m5	37.6	-.42462909
6.	1955m6	39.78	5.6360223
7.	1955m7	42.69	7.0600427
8.	1955m8	42.43	-.61090416
9.	1955m9	44.34	4.4031545
10.	1955m10	42.11	-5.1601962

finregress capm

```
. finregress capm lnr_jsp lnr_cqn lnr_ard lnr_das = lnr_mkt, rfrate(rf) adjustall
Capital asset pricing model
Sample: 1955m2 thru 2019m12                          Number of obs = 779
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
lnr_jsp						
lnr_mkt	.5069003	.0103805	48.83	0.000	.4865548	.5272457
_cons	.0530648	.037391	1.42	0.156	-.0202203	.1263498
lnr_cqn						
lnr_mkt	.5158591	.0157783	32.69	0.000	.4849342	.5467841
_cons	.1449267	.0563692	2.57	0.010	.0344452	.2554083
lnr_ard						
lnr_mkt	.2163326	.0167747	12.90	0.000	.1834548	.2492103
_cons	-.1633412	.0609935	-2.68	0.007	-.2828863	-.0437961
lnr_das						
lnr_mkt	.6014153	.0113724	52.88	0.000	.5791258	.6237047
_cons	.0551297	.038948	1.42	0.157	-.0212069	.1314663

```
Notes: Dependent variables adjusted for risk-free rate rf.
       Independent variable lnr_mkt adjusted for risk-free rate rf.
```

Testing the model

- Do the factors fully explain expected excess returns?

$$r_{it} - r_t^f = \alpha_i + \beta_i(r_t^m - r_t^f) + u_t$$
$$H_0 : \alpha_i = 0 \quad \forall \quad i$$

```
. estat grstest

Gibbons-Ross-Shanken test
H0: All intercept terms are zero

  No. of dependent vars. =       4
No. of independent vars. =       1
     No. of time periods =     779

                 chi2(4) =  11.411
           Prob > chi2 =  0.0223
```

Returning to finsummarize

```
. finsummarize lnr_jsp-lnr_das, rfrate(rf) benchmark(lnr_mkt, adjust) ///
>         statistics(jensen beta)

Financial summary statistics

Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Mean	Std. dev.	Jensen´s alpha	beta
JSP	0.5359	2.0500	0.0531	0.5069
CQN	0.6294	2.3950	0.1449	0.5159
ARD	0.2642	1.8674	-0.1633	0.2163
DAS	0.5559	2.3596	0.0551	0.6014

```
Notes: Variables adjusted for risk-free rate rf.
       Variable lnr_mkt used as benchmark asset and
       adjusted for risk-free rate rf.
```

The cross-section of returns

- $\bar{r}_i = a + \lambda\beta_i + e_i$



The Fama-MacBeth approach

- Estimate β_i for each asset in time-series regressions

$$r_{it} - r_t^f = \alpha_i + \beta_i(r_t^m - r_t^f) + u_{it} \qquad t = 1, \ldots, T$$

- For each time period t, run the cross-sectional regression

$$r_{it} - r_t^f = a_t + \lambda_t \beta_i + e_i \qquad i = 1, \ldots, N$$

- Compute $\widehat{\lambda}$ by the mean of $\{\widehat{\lambda}_t\}$
- Compute standard errors by the variance of $\{\widehat{\lambda}_t\}$

The Fama-MacBeth approach

```
. finregress fmb lnr_jsp-lnr_iron  = lnr_mkt, rfrate(rf) adjust
Fama-MacBeth regression
Sample: 1955m2 thru 2019m12                          Number of obs     = 779
Method: OLS                                          Number of depvars =  24
```

depvar_means	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
beta						
lnr_mkt	.257522	.1283863	2.01	0.045	.0058895	.5091545
_cons	.0061783	.0152296	0.41	0.685	-.0236712	.0360277

```
Dependent variables: lnr_jsp lnr_cqn lnr_ard lnr_das lnr_acme lnr_bat
                     lnr_dune lnr_tyr lnr_glo lnr_spa lnr_wgt ... lnr_iron
Notes: Dependent variables adjusted for risk-free rate rf.
       Independent variable lnr_mkt adjusted for risk-free rate rf.
```

Outline

1 Financial data

2 Portfolios

3 Financial regressions

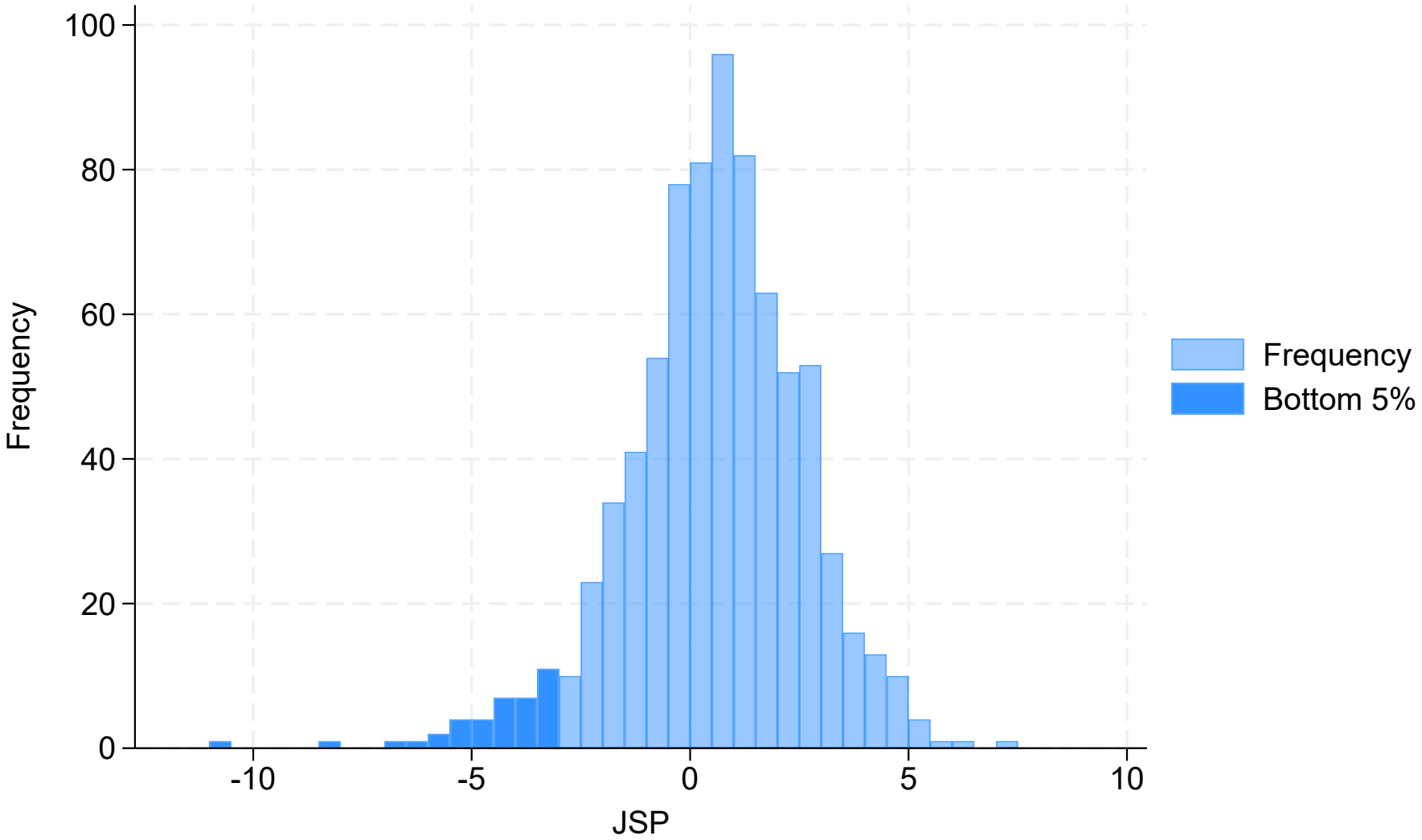
4 Value at Risk

Value at Risk

How much can we lose on a bad day?

- Value at Risk (VaR) puts a loss threshold on bad outcomes
- The 95% VaR is a threshold loss that is only exceeded 5% of the time
- If the 5th percentile of monthly returns is -3%, then on 5% of months, we expect returns to be -3% or worse
- 95% VaR is thus 3%, often reported as a positive number
- Multiply by portfolio size to get a value instead of a percentage

Value at Risk



Value at Risk

- New command `finvalrisk` estimates value at risk

```
. finvalrisk r_jsp r_cqn r_ard r_das p1 p2 p3
Value-at-risk percentiles
Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Empirical	Normal
5%		
JSP	-3.031731	-2.808562
CQN	-3.629369	-3.287768
ARD	-2.758431	-2.796273
DAS	-3.430629	-3.281797
equal weight	-2.260575	-2.250856
fixed	-2.391863	-2.263735
minvar	-2.852225	-2.608141

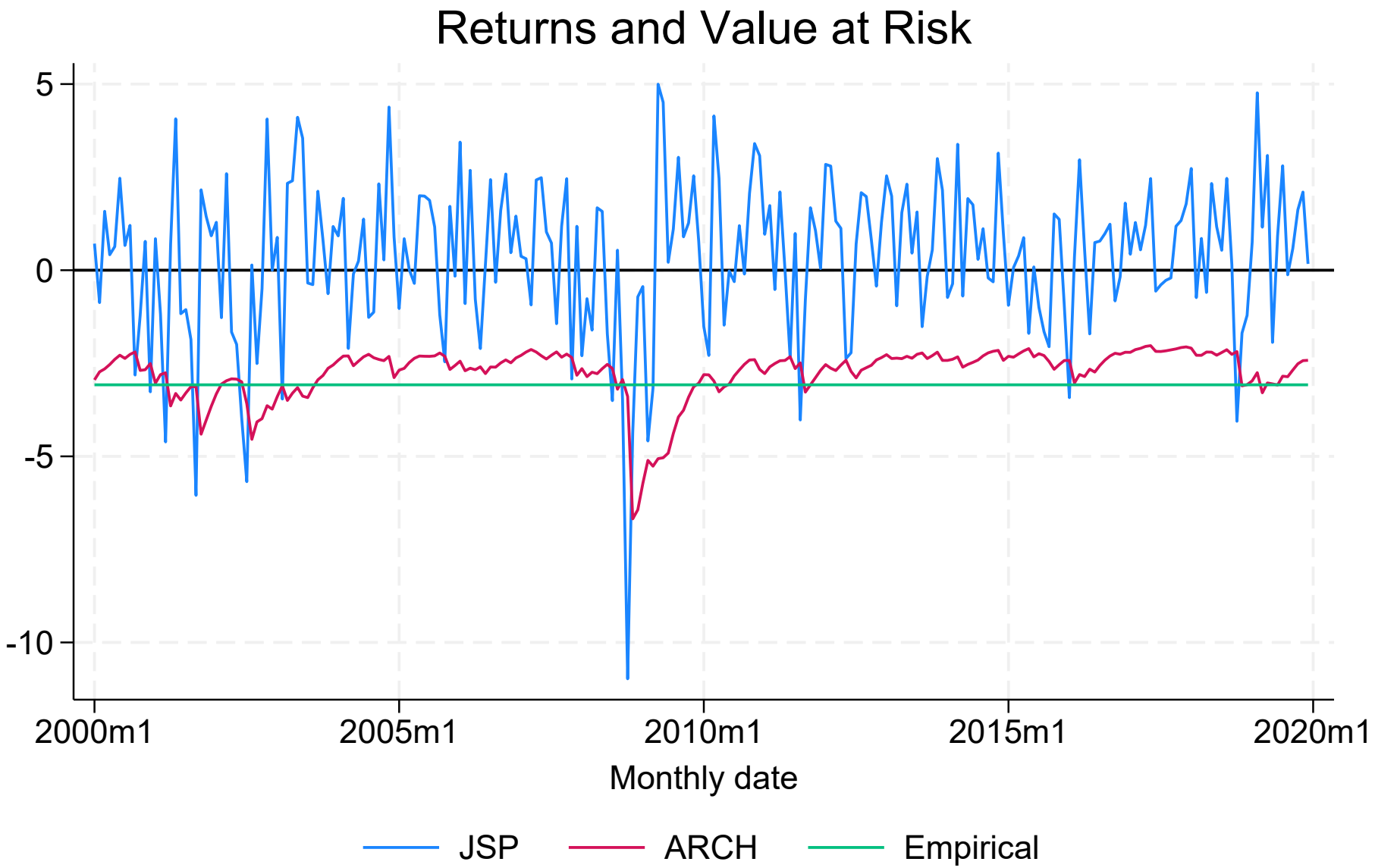
Value at Risk

- Empirical VaR: the q% VaR is the $p = (1 - q/100)$ percentile
- Normal-based VaR

$$r_t = \mu + u_t$$
$$u_t \sim N(0, \sigma^2)$$
$$VaR = \widehat{\mu} + \Phi^{-1}(p)\widehat{\sigma}$$

- Model-based VaR : $VaR = \widehat{\mu}_t + \Phi^{-1}(p)\widehat{\sigma}_t$
  - where $\widehat{\mu}_t$ and $\widehat{\sigma}_t$ are output of a model such as ARIMA or ARCH
  - e.g. $\sigma_t^2 = \omega + \alpha u_{t-1}^2 + \beta\sigma_{t-1}^2$

Value at Risk



Value at Risk

```
. finvalrisk r_jsp r_cqn r_ard r_das p1 p2 p3, model(arch, arch(1) garch(1))
Fitting ARCH models ...
Value-at-risk percentiles
Model: ARCH
Error family: Normal
Number of obs = 779
Sample: 1955m2 thru 2019m12
```

	Empirical	Normal	ARCH historical	ARCH one step
5%				
JSP	-3.031731	-2.808562	-2.733136	-2.31145
CQN	-3.629369	-3.287768	-3.174725	-2.741616
ARD	-2.758431	-2.796273	-2.77472	-2.342388
DAS	-3.430629	-3.281797	-3.160145	-3.029947
equal weight	-2.260575	-2.250856	-2.13826	-1.704063
fixed	-2.391863	-2.263735	-2.156863	-1.67933
minvar	-2.852225	-2.608141	-2.48459	-1.957328

```
Note: One-step-ahead value at risk for 2020m1.
```

A Summing Up

- Generate and manage returns with `finreturns`
- Construct portfolios with `finportoflio`
- Summarize portfolios with `finsummarize`
- Run financial regressions with `finregress`
- Assess value at risk with `finvalrisk`