

Financial Statistics in Stata

David Schenck

Stata

September 3, 2026
2026 UK Stata Conference

Outline

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

The fin suite

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

Outline

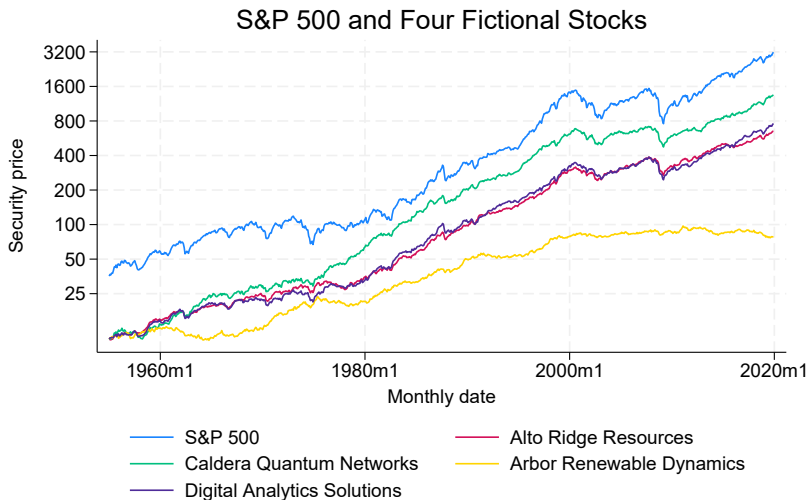
- 1 Financial data
- 2 Portfolios
- 3 Financial regressions
- 4 Value at Risk

Financial data

```
. 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
rf	float	%9.0g		Risk-free rate, monthly
arr	float	%9.0g		Alto Ridge Resources
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
iron	float	%9.0g		Industrial Operations Network
dune	float	%9.0g		Digital Urban Network Enterprise
tyr	float	%9.0g		Tyndale Resources Group

Financial data



From prices to returns

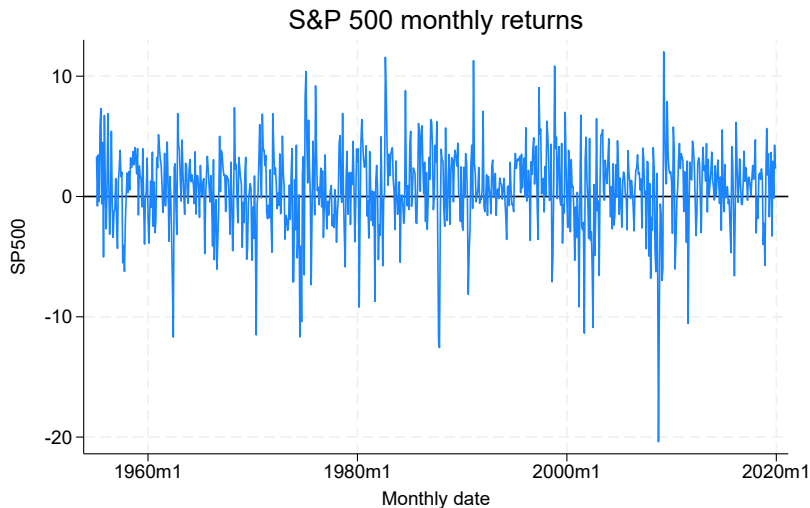
- New command `finreturns` generates returns from prices
 - First difference, simple, log, annualized, log-annualized

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

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

```
. describe r_mkt r_arr r_cqn r_ard r_das
```

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

```
. summarize r_mkt r_arr r_cqn r_ard r_das
```

Variable	Obs	Mean	Std. dev.	Min	Max
r_mkt	779	.6387889	3.458349	-20.39114	12.02171
r_arr	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

Outline

- 1 Financial data
- 2 Portfolios**
- 3 Financial regressions
- 4 Value at Risk

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

Constructing portfolios

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

```
. finportfolio equal  r_arr r_cqn r_ard r_das,      ///
>               generate(p1, label("equal weight"))
```

Equally weighted portfolio

Number of obs = 779

Sample: 1955m2 thru 2019m12

	Weight
ARR	.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_arr 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
ARR	.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_arr 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
ARR	.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_arr 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
ARR	.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_arr r_cqn r_ard r_das p1 p2 p3
```

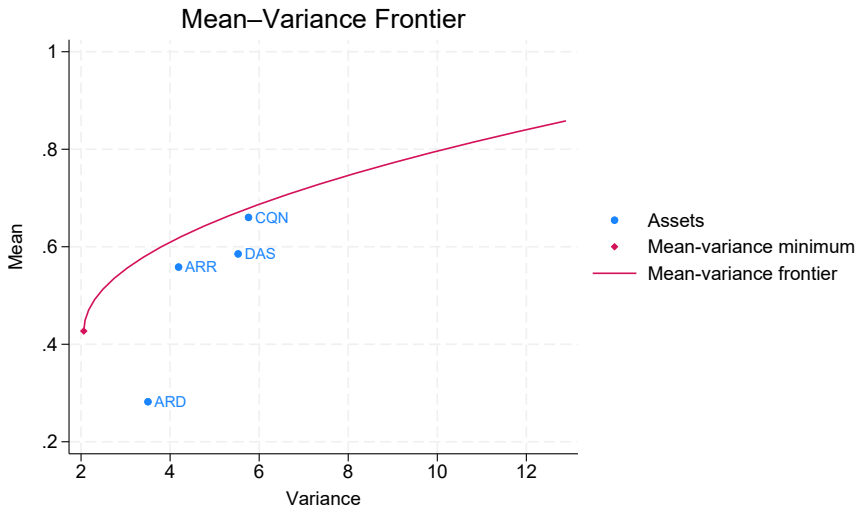
Financial summary statistics

Number of obs = 779

Sample: 1955m2 thru 2019m12

	Mean	Std. dev.	Sharpe ratio
ARR	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

```
. finregress capm r_arr r_cqn r_ard r_das = r_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]	
r_arr	r_mkt	.5068281	.010775	47.04	0.000	.4857095	.5279468
	_cons	.2326441	.0384613	6.05	0.000	.1572613	.3080269
r_cqn	r_mkt	.5184283	.0164889	31.44	0.000	.4861106	.5507461
	_cons	.3271305	.05771	5.67	0.000	.2140209	.4402401
r_ard	r_mkt	.2148353	.0172686	12.44	0.000	.1809895	.2486811
	_cons	.1417603	.0630914	2.25	0.025	.0181034	.2654172
r_das	r_mkt	.6019268	.0116667	51.59	0.000	.5790605	.6247931
	_cons	.1991861	.0402332	4.95	0.000	.1203305	.2780418

Notes: Dependent variables adjusted for risk-free rate rf.

Independent variable r_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) = 127.321

Prob > chi2 = 0.0000

Returning to finsummarize

```
. finsummarize r_arr-r_das, rfrate(rf) benchmark(r_mkt, adjust) ///
>          statistics(jensen beta)
```

Financial summary statistics

Number of obs = 779

Sample: 1955m2 thru 2019m12

	Mean	Std. dev.	Jensen's alpha	beta
ARR	0.5583	2.0469	0.2326	0.5068
CQN	0.6602	2.4002	0.3271	0.5184
ARD	0.2820	1.8715	0.1418	0.2148
DAS	0.5852	2.3510	0.1992	0.6019

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

The cross-section of returns

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



The Fama-MacBeth approach

```
. finreturns arr-tks, multiply(100) simple(r) numbered nopreview
(all returns multiplied by 100)
```

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

```
. finregress fmb r1-r24 = r_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						
r_mkt	.3720936	.1267233	2.94	0.003	.1237205	.6204668
_cons	.3610551	.0192755	18.73	0.000	.3232759	.3988343

Dependent variables: r1 r2 r3 r4 r5 r6 r7 r8 r9 r10 r11 r12 r13 r14 r15 r16
r17 r18 r19 r20 r21 r22 r23 r24

Notes: Dependent variables adjusted for risk-free rate rf.

Independent variable r_mkt adjusted for risk-free rate rf.

The Fama-MacBeth approach

- Estimate β_i for each asset in time-series regressions
- For each time period t , run the cross-sectional regression

$$r_{it} = a_t + \lambda_t \beta_i + e_i \quad i = 1, \dots, N$$

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

Outline

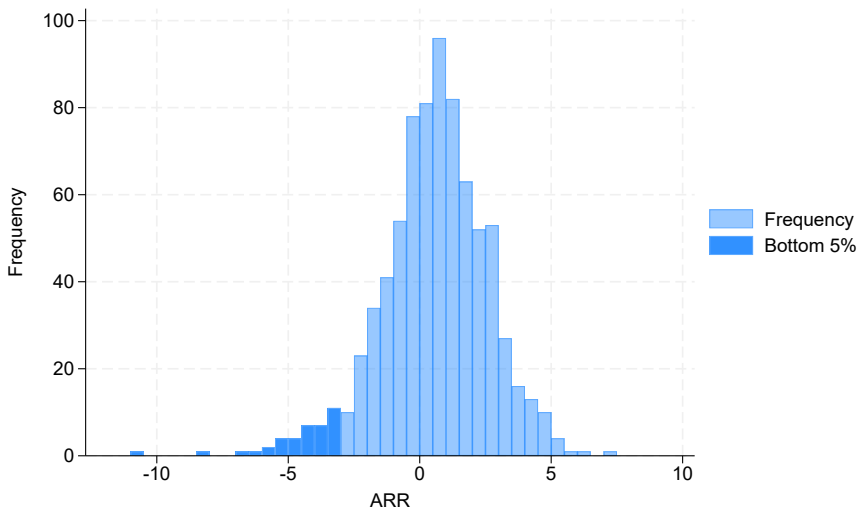
- 1 Financial data
- 2 Portfolios
- 3 Financial regressions
- 4 Value at Risk**

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_arr r_cqn r_ard r_das p1 p2 p3
```

Value-at-risk percentiles

Number of obs = 779

Sample: 1955m2 thru 2019m12

		Empirical	Normal
5%			
	ARR	-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

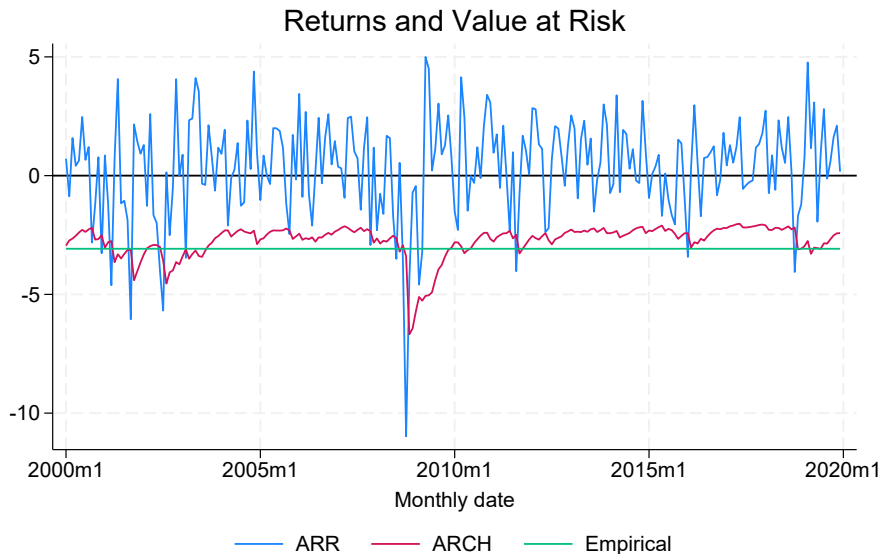
$$r_t = \mu + u_t$$

$$u_t \sim N(0, \sigma^2)$$

$$VaR = \hat{\mu} + \Phi^{-1}(p)\hat{\sigma}$$

- Model-based VaR : $VaR = \hat{\mu}_t + \Phi^{-1}(p)\hat{\sigma}_t$
 - where $\hat{\mu}_t$ and $\hat{\sigma}_t$ 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_arr 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

		ARCH			
		Empirical	Normal	historical	one step
5%					
	ARR	-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`