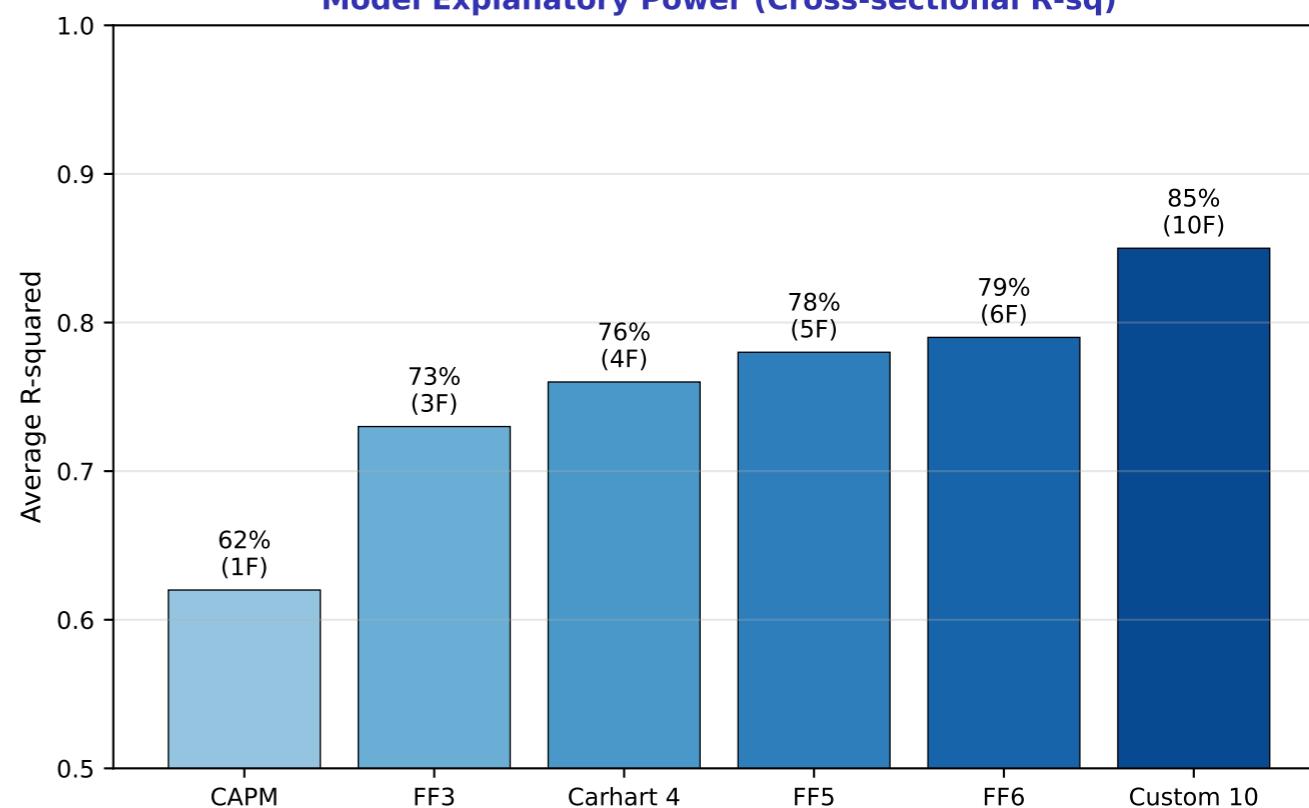


# Multi-Factor Models: Comprehensive Risk Decomposition

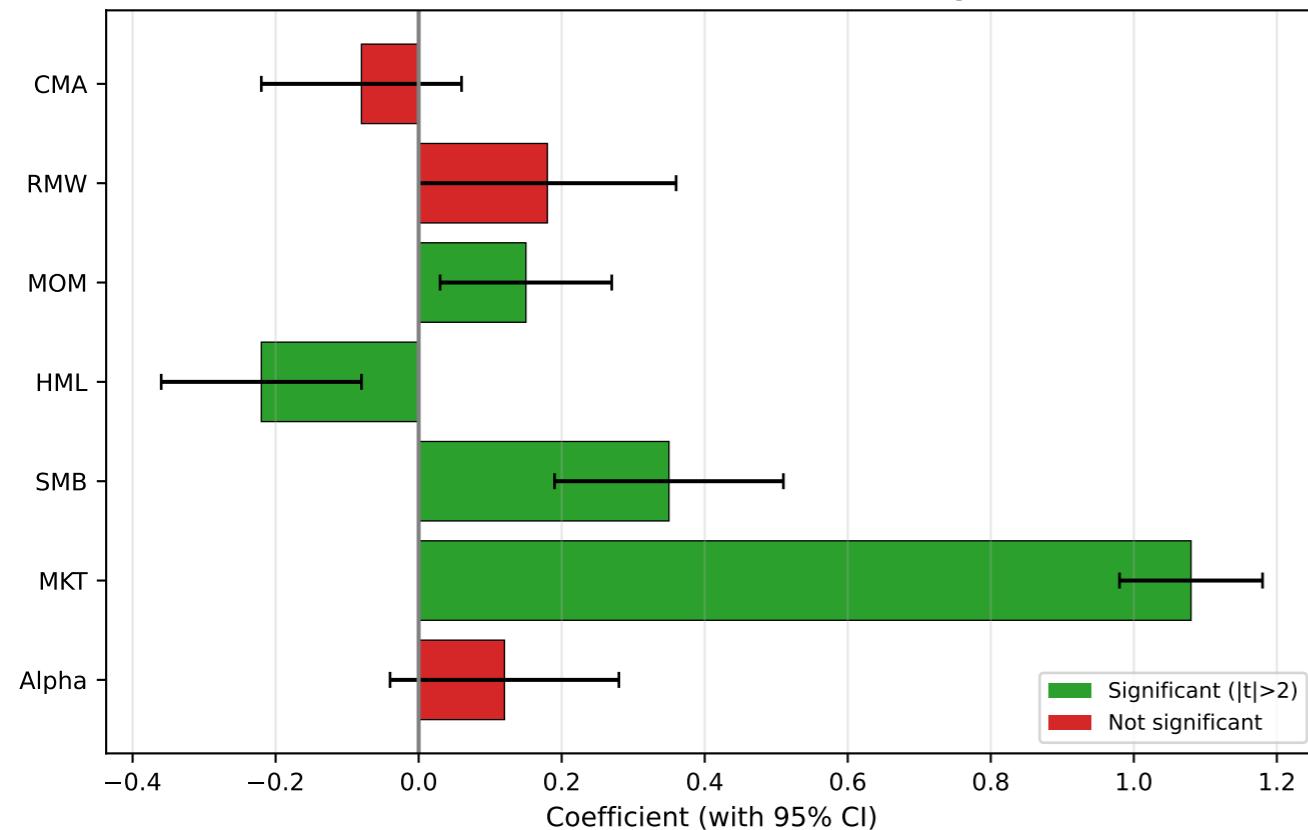
## The Factor Zoo

THE FACTOR ZOO	
Academic Factors (Well-Established)	
Market (MKT)	: Equity premium
Size (SMB)	: Small cap premium
Value (HML)	: Value vs growth
Momentum (MOM)	: Winners keep winning
Profitability	: Profitable > unprofitable
Investment	: Conservative > aggressive
Quality	: Low debt, stable earnings
Industry Factors (Practitioner)	
BAB	: Betting Against Beta
QMJ	: Quality Minus Junk
LIQ	: Liquidity risk premium
VOL	: Volatility factor
TERM	: Term structure
CREDIT	: Credit spread
THE PROBLEM: Factor proliferation	
-	400+ published factors
-	Many don't replicate
-	Data mining concerns
-	Use skeptically!

## Model Explanatory Power (Cross-sectional R-sq)



## Fama-French 6-Factor Model Output



## Python Implementation

```

Multi-Factor Model in sklearn

from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
import statsmodels.api as sm

# Prepare factor data
factors = ['MKT', 'SMB', 'HML', 'MOM', 'RMW', 'CMA']
X = ff_data[factors]
y = stock_excess_returns

# Option 1: sklearn (simple)
model = LinearRegression()
model.fit(X, y)
print(f"Alpha: {model.intercept_:.4f}")
print(f"Betas: {dict(zip(factors, model.coef_))}")
print(f"R-squared: {r2_score(y, model.predict(X)):.4f}")

# Option 2: statsmodels (with t-stats)
X_sm = sm.add_constant(X) # Add intercept
model_sm = sm.OLS(y, X_sm).fit()
print(model_sm.summary())

# Key outputs from statsmodels:
# - Coefficients with standard errors
# - t-statistics and p-values
# - R-squared and Adjusted R-squared
# - F-statistic for overall significance

```