CAPM是由William Sharpe于1964创建的一种模型，它根据市场回报率和资产与市场回报率的线性关系来估计资产的回报率。这种线性关系的斜率就是股票的贝塔系数，贝塔系数可以被视为股票对市场的敏感度，或者是它对市场的风险度量。贝塔系数是指一项资产的贝塔系数，它是将该资产的回报率与市场回报率进行回归得出的。它捕捉了资产和市场之间的线性关系。

CAPM是一个很好的工具，可以探索一个可重复的流程，用于建模或回归我们的投资组合在市场上的回报。

1. CAPM和市场回报：我们的第一步是选择哪种资产作为市场回报的代理。文中选取了SPY ETF作为对象。我们首先输入SPY的价格，计算每月的回报，并保存为market\_returns\_xts.。并确保投资组合回报率观测值的数量等于市场回报率观测值的数量。最后得到各个时期的市场回报率。
2. 计算CAPM的β：投资组合的β等于投资组合收益和市场收益的协方差除以市场收益的方差。使用cov()计算投资组合和市场收益的协方差，用var()计算市场收益的方差。此外，我们还可以通过找到各个资产的β值进行加权平均计算出投资组合的β。

之后，使用lm（）函数根据市场回报率对每个资产回报率进行回归。但是当组合中资产数量过多时，我们需要用map()函数进行回归。以此，我们可以得到一个表格，但是得到的数据的β值被储存为s3对象，并不适合演示。

因此，我们可以使用broom包中的tidy()函数清理结果，并使用mutate()和map()组合将该函数应用到模型中。

此后，为了更可靠并且更加直观，我们用unnest()函数来规范数据栏。

1. 在xts world中计算β：我们从xts开始，并使用PerformanceAnalytics包中内置的capm.beta（）函数。这个函数有两个参数：我们希望计算其beta的投资组合的回报率和市场回报率。
2. 在tidyverse中计算β：在tidyverse包中查找和显示portfolio beta时，需要在market returns对象上回归portfolio returns对象，然后使用broom函数清理结果。
3. 在tidyquant world中计算β：我们使用tq\_performance（）将capm.beta（）应用于回归对象
4. 用ggplot将CAPM可视化：
5. 投资组合与市场的散点图：创建一个散点图，X轴表示市场收益，Y轴表示投资组合收益。然后，我们可以用geom\_smooth（method=“lm”，se=false，…）添加一个简单的回归线。
6. 用ggplot做回归线散点图：在a中，我们创建的回归线的斜率就是投资组合的β。为了证实这一结论，我们可以在散射图中添加一条直线，其斜率等于beta计算值，y截距等于我在beta-dplyr-u byhand对象中标记为alpha的值。用geom\_abline()添加线。
7. 估计β值的回归线散布：我们可以去确认a和b中的两条线是重合的，由于b中我们手动添加的是一条射线，所以它应从回归线的终点开始显现。
8. 扩大数据量：之后代码块将从lm（returns~market\_returns\_tidy$returns…）的模型结果开始，然后用augment()扩展原始数据集，存储在名为.fitted的新列中。再用ggplot看拟合的数据与实际返回值的匹配程度
9. 用highcharter可视化CAPM：
10. 首先用portfolio\_model\_augmented$returns,将投资组合回报进行回归，再用portfolio\_model\_augmented$mkt\_rtns.对市场回报进行回归。
11. 之后，我们用hc\_tooltip(formatter = JS("function(){return ('port return:' + this.y + ' <br> mkt return: ' + this.x + ' <br> date: ' +this.point.date)}")).工具来提取并演示变量，让每个点上都可以现实时间和具体数据
12. 最后，我们需要添加一条回归线，为此，我们需要为highcharter提供x和y坐标，们有拟合回归线的x和y坐标，因为我们用augment（）函数添加了它们。x是市场收益，y是拟合值。我们使用hc\_add\_series（portfolio\_model\_augmented，type=“line”，hcaes（x=mkt\_rtns，y=.fitted））添加回归线。这样所得到的图与之前用ggplot做出来的图几乎一致，但这样做出来的图更具有交互性（interactive）

9、用Shiny app做CAPM