[{"metadata":{"trusted":true},"cell\_type":"code","source":"import csv #this calls the comma separated values file library\nimport pandas as pd","execution\_count":2,"outputs":[]},{"metadata":{"trusted":true},"cell\_type":"code","source":"df = pd.read\_csv(\"cropmixdata.csv\") #this saves it under the name 'df'","execution\_count":3,"outputs":[]},{"metadata":{"trusted":true},"cell\_type":"code","source":"df","execution\_count":4,"outputs":[{"data":{"text/html":"

\n\n\n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n \n

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Land Use** | **Water Need (mm/year)** | **Revenue ($/acre)** |
| **0** | Alfalfa | 1200.000000 | 675 |
| **1** | Corn | 650.000000 | 699 |
| **2** | Cotton | 1000.000000 | 470 |
| **3** | Sorghum | 550.000000 | 244 |
| **4** | Soybean | 575.000000 | 561 |
| **5** | Wheat | 550.000000 | 205 |
| **6** | Yearling Cattle | 0.000031 | 869 |

\n

","text/plain":" Land Use Water Need (mm/year) Revenue ($/acre)\n0 Alfalfa 1200.000000 675\n1 Corn 650.000000 699\n2 Cotton 1000.000000 470\n3 Sorghum 550.000000 244\n4 Soybean 575.000000 561\n5 Wheat 550.000000 205\n6 Yearling Cattle 0.000031 869"},"execution\_count":4,"metadata":{},"output\_type":"execute\_result"}]},{"metadata":{"trusted":true},"cell\_type":"code","source":"import altair as alt","execution\_count":5,"outputs":[]},{"metadata":{"trusted":true},"cell\_type":"code","source":"#BCG Matrix maker\nalt \\\n .Chart(df) \\\n .mark\_circle(size=200) \\\n .encode(#allows me to specify details of the graph\n x = alt.X('Revenue ($/acre):Q', scale = alt.Scale(zero = False), title = 'Revenue ($/acre)'), #data used for x variable\n y = alt.Y('Water Need (mm/year):Q', scale = alt.Scale(zero = False)), #data used for y\n color = 'Land Use', #color codes each data point by company name\n) .properties( #makes the chart easier to see in these sizes\n width = 720,\n height = 480\n)","execution\_count":8,"outputs":[{"data":{"text/html":"\n\n","text/plain":"alt.Chart(...)"},"execution\_count":8,"metadata":{},"output\_type":"execute\_result"}]},{"metadata":{"trusted":true},"cell\_type":"code","source":"","execution\_count":null,"outputs":[]}]