

1978 年至 2018 年中国居民人均可支配收入和消费支出的数学模型

吴承宇 20230616

刘渭渊 20230610

叶泽宇 20230621

吴可悠 20230617

2023.12.31

摘要

改革开放以来,我国经济增长,人民生活水平不断提高.人均收入水平是一个重要的宏观经济指标,也是衡量一个国家经济发展水平的重要指标.本文通过建立线性回归模型,分析了从 1978 年至 2018 年中国居民人均可支配收入和支出的数据,并且预测了未来的发展趋势.

关键词

数学模型 中国经济 居民收入 居民支出 线性回归

1 问题背景

改革开放以来,中国经济发展迅速,居民收入和支出也随之增长.本文将对 1978 年至 2018 年中国居民人均可支配收入和支出的数据进行分析,建立数学模型,预测未来的发展趋势.

改革开放以来中国经济发展的客观因素主要有:对外开放程度的扩大,人口红利的释放,以及自 2001 年加入世界贸易组织后,中国经济与世界经济的深度融合.

中国共产党的领导对于改革开放的成功起到了决定性的作用.中国共产党的领导使得中国能够在改革开放的过程中保持社会稳定,从而使得改革开放能够顺利进行.

本文通过建立线性回归模型,分析了从 1978 年至 2018 年中国居民人均可支配收入和支出的数据,并且对于城镇和乡村进行线性比较.本文的具体安排如下:

- 第 2 节,对于数据进行了分析,通过绘制散点图,分析了数据的分布规律.
- 第 3 节,通过建立线性回归模型,对于数据进行了分析,并且预测了未来的发展趋势.
- 第 4 节,对于题目所给出的问题进行了解答.
- 第 5 节,对于城镇和乡村的数据进行了分析,并且对于城镇和乡村的数据进行了线性比较.

- 第 6 节, 对于模型的优缺点进行了分析, 并且对于模型的改进进行了讨论.

2 数据分析

通过对于数据绘制散点图 (见图 1, 图 2), 可以发现数据的分布规律.

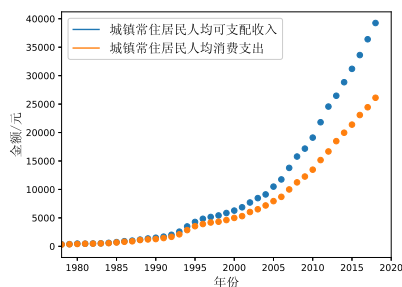


图 1: 城镇可支配收入和支出散点图

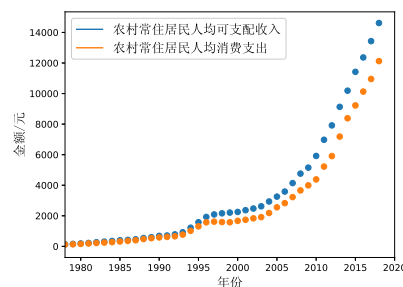


图 2: 乡村可支配收入和支出散点图

通过图 1 和图 2 可以发现, 从 1978 年至 2018 年, 中国居民人均可支配收入和支出主要呈现指数增长的趋势.

整体而言, 收入增长速度大于支出增长速度, 且城镇的差值增长速度大于乡村.

同时, 将图 1 和图 2 叠加在一起 (见图 3), 可以发现城镇和乡村的数据分布规律基本相同, 但是城镇的增长速度明显快于乡村, 并且随着时间的增长, 差距越来越大.

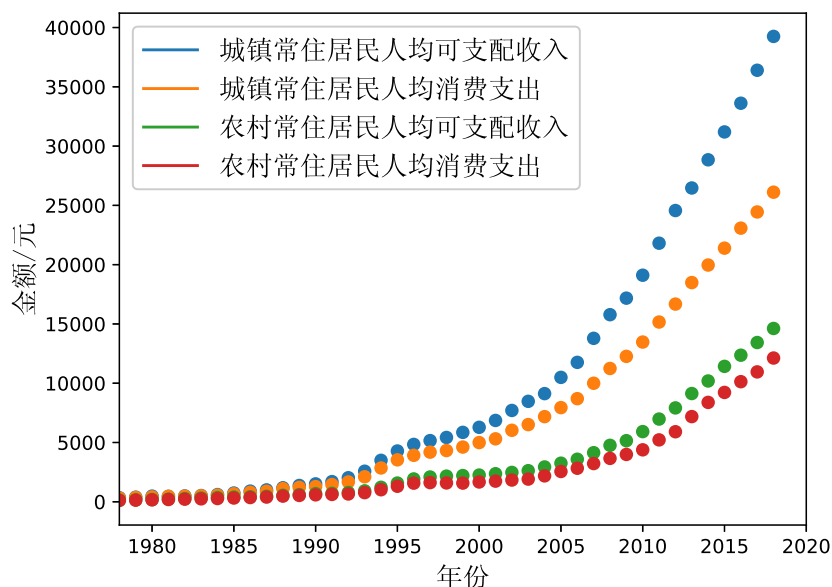


图 3: 可支配收入和支出散点图

从图 3 中可以发现, 城镇人均可支配收入和消费支出的数值首先与乡村持平, 随着时间的推移, 城镇的数值逐渐超过乡村, 并且差距越来越大.

3 模型建立与分析

3.1 假设条件

- 假设居民人均可支配收入和支出的增长速度为指数增长.
- 假设没有其他因素影响居民人均可支配收入和支出的增长速度.
- 假设中国在未来的发展中, 经济发展的速度不会发生剧烈的变化.

3.2 模型建立

笔者使用线性回归分析的方法, 建立了居民人均可支配收入和支出的数学模型 (公式 1).

通过如下的模型, 分别对于城镇人均可支配收入, 城镇人均消费支出, 乡村人均可支配收入, 乡村人均消费支出进行线性回归分析:

$$y = \theta_0 e^{\theta_1(x-1978)} \quad (1)$$

通过自然对数 e 的幂函数, 可以将指数函数转化为线性函数, 从而使用线性回归分析的方法进行分析.

通过使用 Python 的 `sklearn` 库, 笔者得到了如下的结果:

表 1: 线性回归分析结果

数据	θ_0	θ_1	loss	误差率	表达式
城镇可支配收入	359.939478	0.124215	0.98911883	1.088%	$y = 359.939478e^{0.124215(x-1978)}$
城镇消费支出	345.872895	0.114970	0.98795625	1.204%	$y = 345.872895e^{0.114970(x-1978)}$
乡村可支配收入	173.015491	0.112898	0.99034022	1.066%	$y = 173.015491e^{0.112898(x-1978)}$
乡村消费支出	145.093811	0.110557	0.98983601	1.016%	$y = 145.093811e^{0.110557(x-1978)}$

通过对于数据进行线性回归分析, 可以发现, 误差率都在 1% 以左右, 因此模型的拟合效果较好.

3.3 模型分析

通过对于模型的分析, 可以发现, 从 1978 年至 2018 年, 中国居民人均可支配收入和支出主要呈现指数增长的趋势.

整体而言, 收入增长速度大于支出增长速度, 且城镇的差值增长速度大于乡村. 从数据上, 城镇的两个数据的系数值 (θ_0) 都约等于乡村的两个数据的系数值的 2 倍. 同时, 城镇的两个数据的指数值 (θ_1) 都约等于乡村的两个数据的指数值.

通过对于数据模型的建立, 笔者发现和先前观察散点图的结果大致相同, 但是对于起始的数据分析由于散点图的缩放倍率大, 因此难以准确分析.

3.4 模型预测

假设该发展趋势不变的情况下, 未来的发展趋势将会如图所示 (1980 – 2040).

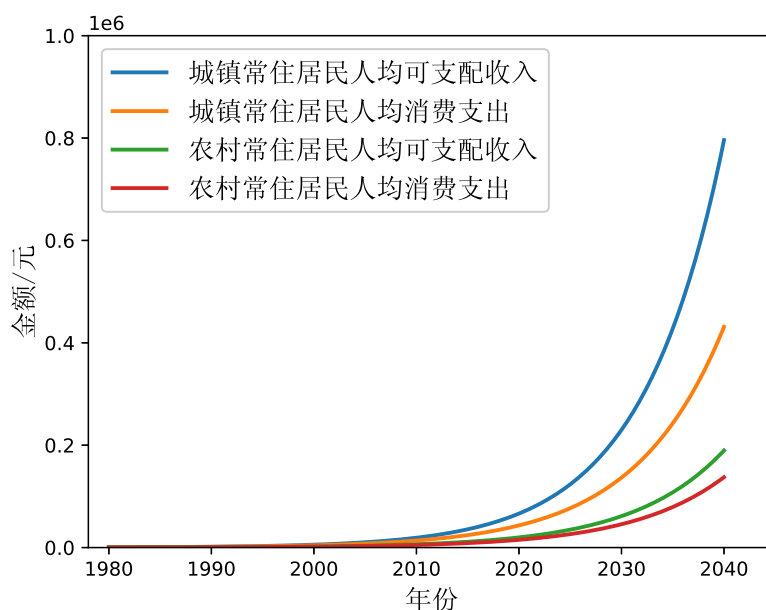


图 4: 未来发展趋势

通过对于图 4 的分析, 可以发现, 未来的发展趋势将会如下所示:

- 城镇人均可支配收入和消费支出的差值将会越来越大.
- 乡村人均可支配收入和消费支出的差值将会越来越大.
- 城镇人均可支配收入和消费支出的差值将会大于乡村人均可支配收入和消费支出的差值.

根据模型, 2040 年各项数据均会达到 2020 年的 10 倍左右.

4 问题解答

4.1 问题一

问题 1: 那么 5 年后 (2023 年)、10 年后 (2028 年) 和 20 年后 (2038 年), 我国居民的人均可支配收入及消费支出分别会达到什么水平?

5 年后 (2023 年), 10 年后 (2028 年) 和 20 年后 (2038 年), 我国居民的人均可支配收入及消费支出分别会达到如下表 2 水平 (单位: 元).

表 2: 问题 1 未来发展趋势

年份	城镇可支配收入	城镇消费支出	乡村可支配收入	乡村消费支出
2023	96337.3045	61066.6611	27827.8093	21003.5239
2028	179276.7357	108507.1504	48936.6133	36505.9487
2038	620844.4837	342583.7225	151336.5031	110282.4968

4.2 问题二

问题 2: 试评价问题 (1) 中建立的模型效果.

首先, 从数据角度分析, 表 1 中显示, 拟合数据误差率都在 1% 以左右, 因此模型的拟合效果较好.

其次, 图 4 和散点图 (见图 3) 的结果基本相同.

最后, 从模型的角度分析, 通过对于模型的分析, 可以发现, 从 1978 年至 2018 年, 中国居民人均可支配收入和支出主要呈现指数增长的趋势.

但是, 由于指数函数模型的特殊性, 到后期时间越长, 增长速度逐渐呈现爆炸性增长, 因此模型的预测结果可能会出现较大的误差.

4.3 问题三

问题 3: 一位同学尝试用二次函数

$$h(x) = \theta_1 x^2 + \theta_0 \quad (2)$$

来拟合 1978 - 1982 这 5 年的城镇常住居民人均收入可支配, 用这个假设函数得到的 2038 年城镇常住居民人均收入可支配为多少? 与 (1) 的效果相比哪一个更好?

首先, 根据要求的二次函数拟合模型, 通过 Python 的 `sklearn` 库, 笔者得到了如下的结果:

$$h(x) = 384.786207 + 7.902299 * (x - 1978)^2$$

其中误差 $loss = 0.65037356$, 误差率为 0.715%.

通过预测, 得到 2038 年城镇常住居民人均收入可支配为 $h(2038) = 28833.062607$ 元. 然而, 根据数据, 从 2014 年开始, 城镇常住居民人均收入可支配已经超过了 28833 元.

因此, 二次函数模型的预测结果不及笔者所建立模型准确.

5 数据特性比较

5.1 城乡各自收入支出数据分析

通过对城乡各自数据以收入为横轴, 支出为纵轴进行散点图绘制 (见图 5 和图 6), 可以发现, 城镇和乡村的数据分布规律基本相同. 二者比例基本都呈现线性增长趋势

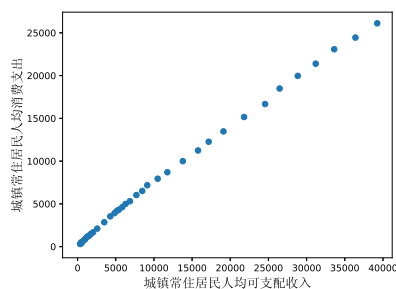


图 5: 城镇各自收入支出数据分析

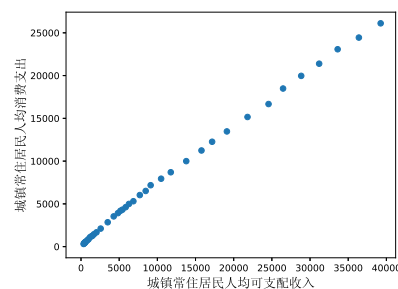


图 6: 乡村各自收入支出数据分析

5.2 城乡收入支出数据比较

通过以城镇收入 (支出) 的数据为横轴, 乡村收入 (支出) 的数据为纵轴进行散点图绘制 (见图 7 和图 8), 可以发现, 二者的趋势也基本相同, 整体多呈现二次函数的趋势.

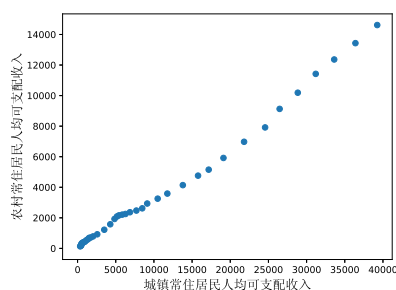


图 7: 城乡可支配收入比较

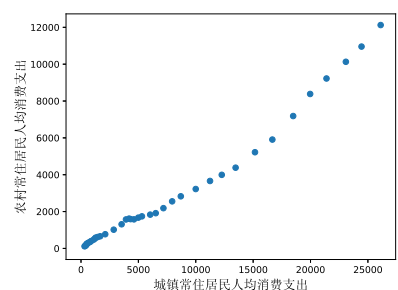


图 8: 城乡消费支出比较

5.3 综合分析总结

总体而言, 城乡的数据趋势基本同步, 但是城镇的数据趋势相较于乡村的数据趋势更为明显.

乡村由于其固有的产业结构特性, 发展的势头不如城镇, 因此应当着眼于对乡村的产业转型, 加快建设产业化, 从而提高乡村的发展速度.

由于笔者经历有限, 不再建立回归模型进行分析. 但是, 根据散点图也可以从中略知一二.

6 模型优缺点分析

6.1 模型优点

- 通过建立指数函数的回归模型, 笔者发现该模型较好的拟合了数据, 并且预测了未来的发展趋势. 同时, 以自然常数 e 为底数的幂函数, 既能使计算简单, 也可以相对完好的进行拟合分析和数据预测, 较符合经济发展的自然规律和实际情况.
- 通过使用 Python 的 `sklearn` 库进行线性回归分析, 笔者发现该模型的误差率都在 1% 以左右, 因此模型的拟合效果较好. 同时, 模型建立也较为简单, 且计算量较小.
- 通过 `matplotlib` 库绘制散点图, 可以直观的看出数据的分布规律, 从而对于数据进行分析.

6.2 模型缺点

- 指数函数模型的增长速率随着时间的增长呈现爆炸性增长, 因此模型的预测结果可能会出现较大的误差.
- 该模型基于 1978 – 2018 年的数据进行分析, 没有考虑到其他因素的影响. 例如, 2019 年底开始的新冠肺炎疫情对于经济的影响, 以及从 2020 年前后开始的国际形势的变化等等.
- 该模型没有考虑到不同产业结构对于经济发展的影响, 例如, 部分城市的经济发展主要依靠土地财政, 部分城市的经济发展依靠工业, 部分城市的经济发展依靠服务业等等. 不同产业的发展速度和能力不同, 因此对于经济发展的影响也不同.
- 人均收入水平作为一个宏观指标, 并不能完全反映一个国家的经济发展水平. 同时, 对于 CPI, GNP, GDP 等指标的分析也是十分重要的.

6.3 模型改进

- 可以通过建立多项式回归模型, 从而减少指数函数模型的爆炸性增长的问题.
- 可以通过建立多元回归模型, 从而考虑到其他因素的影响.
- 可以通过建立多元回归模型, 从而考虑到不同产业结构对于经济发展的影响.
- 可以对 1978 – 2000 年和 2000 – 2018 年的数据分别进行分析, 从而考虑到不同时间段的影响.
- 可以对于 CPI, GNP, GDP 等指标进行分析, 从而考虑到不同指标的影响.

鸣谢

感谢洪伟智老师和鲍可宇老师在数学建模课程中的指导和帮助. 感谢 Python 和 \LaTeX 以及 VS Code 等开源软件.

感谢 GitHub 提供的代码托管服务.

感谢 `xelatex`, `bibtex` 等工具提供的支持.

感谢一直陪伴我写论文的布蕾脆脆奶芙.

参考文献

- [1] 刘伟 and 张辉. 中国经济增长中的产业结构变迁和技术进步. 经济研究, (11):12, 2008.
- [2] 刘来福, 黄海洋, and 曾文艺. 数学模型与数学建模. 数学模型与数学建模, 2009.
- [3] 华罗庚 and 王元. 数学模型选谈. 数学模型选谈, 2011.
- [4] 周星 and 克居正. 男生追女生的数学模型. 数学的实践与认识, 24(12):8, 2012.
- [5] 张长生 and 刘殿兰. 改革开放以来中国居民人均收入增长与 gdp 增长的相关性分析及实现两者同步翻番的侧重点. In 市场经济与增长质量——2013 年岭南经济论坛暨广东经济学会年会论文集, 2013.

[4] [5] [3] [1] [2]

附录

A 代码

A.1 散点图绘制

```
1 import matplotlib.pyplot as plt
2 from matplotlib.font_manager import FontProperties
3 import numpy as np
4 import pandas as pd
5 import os.path as path
6
7 data = pd.read_csv(path.join(path.dirname(__file__), "./data.csv"), encoding="utf-8")
8
9 # Create 4 plots (scatter) of the data (total 4) (line 0 as the x-axis, line :i: as the y-
   axis)
10
11 font = FontProperties(fname=path.join(path.dirname(__file__), "./SimSun.ttf"), size=14)
12
13 total = 0
14
15
16 def plotbase(fr: int, to: int):
17     global total
18     total += 1
19     for i in range(fr, to):
20         plt.plot(1, i, label=data.columns[i])
21         plt.scatter(data.iloc[:, 0], data.iloc[:, i])
22         plt.xlabel(data.columns[0], fontproperties=font)
23         plt.ylabel("金额/元", fontproperties=font)
24         plt.xlim(1978, 2020)
25     plt.legend(prop=font)
26     plt.savefig(
```



```

27         path.join(
28             path.dirname(__file__),
29             f"../figures/plot{str(total)}.eps",
30         )
31     )
32     plt.close()
33
34
35 plotbase(1, 5)
36 plotbase(1, 3)
37 plotbase(3, 5)

```

A.2 线性回归分析

```

1  import pandas as pd
2  import numpy as np
3  import os.path as path
4  from sklearn.linear_model import LinearRegression
5
6  data = pd.read_csv(path.join(path.dirname(__file__), "../data.csv"), encoding="utf-8")
7
8  # replace the first column of each row with the (year - 1978)
9
10 data['年份'] = data['年份'] - 1978
11
12 def regression(line: int):
13     x = data.iloc[:, 0].values.reshape(-1, 1)
14     y = data.iloc[:, line].values.reshape(-1, 1)
15
16     y_log = np.log(y)
17
18     model = LinearRegression()
19     model.fit(x, y_log)
20
21     b = model.coef_[0][0]
22     a = np.exp(model.intercept_[0])
23
24     print(f'y = {a:.6f} * e^({b:.6f} * x)')
25
26     loss = model.score(x, y_log)
27
28     print(f'loss = {loss:.8f}')
29
30 for i in range(1, 5):
31     regression(i)

```

A.3 模型预测

```

1  import math
2  import numpy as np
3  import pandas as pd
4  import os.path as path
5  import matplotlib.pyplot as plt
6  from matplotlib.font_manager import FontProperties
7
8
9  def fn1(x):
10     return 359.939478 * math.exp(0.124215 * (x - 1978))
11
12
13  def fn2(x):
14     return 345.872895 * math.exp(0.114970 * (x - 1978))
15
16
17  def fn3(x):
18     return 173.015491 * math.exp(0.112898 * (x - 1978))
19
20
21  def fn4(x):
22     return 145.093811 * math.exp(0.110557 * (x - 1978))
23
24
25  x = np.linspace(1980, 2040, 1000)
26  y1 = np.array([fn1(i) for i in x])
27  y2 = np.array([fn2(i) for i in x])
28  y3 = np.array([fn3(i) for i in x])
29  y4 = np.array([fn4(i) for i in x])
30
31  for i in [2023, 2028, 2038]:
32     print(f"{i} & ${fn1(i):.4f}$ & ${fn2(i):.4f}$ & ${fn3(i):.4f}$ & ${fn4(i):.4f}$ \\\\")
33
34  font = FontProperties(fname=path.join(path.dirname(__file__), "./SimSun.ttf"), size=14)
35
36  title = ["年份", "城镇常住居民人均可支配收入", "城镇常住居民人均消费支出", "农村常住居民人均可支配收入", "农村常住居民人均消费支出"]
37
38  plt.plot(x, y1, linewidth=2, linestyle="--", label=title[1])
39  plt.plot(x, y2, linewidth=2, linestyle="--", label=title[2])
40  plt.plot(x, y3, linewidth=2, linestyle="--", label=title[3])
41  plt.plot(x, y4, linewidth=2, linestyle="--", label=title[4])
42
43  plt.xlabel(title[0], fontproperties=font)
44  plt.ylabel("金额/元", fontproperties=font)
45
46  plt.xlim(1978, 2045)
47  plt.ylim(0, 1000000)
48

```

```

49 plt.legend(prop=font)
50
51 plt.savefig(
52     path.join(
53         path.dirname(__file__),
54         f"../figures/result1.eps",
55     )
56 )

```

A.4 问题 3 二次函数模型

```

1  import pandas as pd
2  import numpy as np
3  from sklearn.linear_model import LinearRegression
4
5  df = pd.DataFrame({
6      'year': [0, 1, 2, 3, 4],
7      'city_income': [343, 387, 478, 458, 495],
8  })
9
10 x = df['year'].values.reshape(-1, 1)
11 y = df['city_income'].values.reshape(-1, 1)
12 x_pow = np.power(x, 2)
13
14 '''
15 function:  $h(x) = \theta_1 x^2 + \theta_0$ 
16 '''
17
18 model = LinearRegression()
19 model.fit(x_pow, y)
20
21 b = model.coef_[0][0]
22 a = model.intercept_[0]
23
24 print(f'y = {a:.6f} + {b:.6f} * x^2')
25
26 loss = model.score(x_pow, y)
27
28 print(f'loss = {loss:.8f}')

```

A.5 横向比较分析

```

1  import matplotlib.pyplot as plt
2  from matplotlib.font_manager import FontProperties
3  import numpy as np
4  import pandas as pd
5  import os.path as path
6

```

```

7 data = pd.read_csv(path.join(path.dirname(__file__), "../data.csv"), encoding="utf-8")
8
9 # Create 4 plots (scatter) of the data (total 4) (line 0 as the x-axis, line :i: as the y-
  axis)
10
11 font = FontProperties(fname=path.join(path.dirname(__file__), "../SimSun.ttf"), size=14)
12
13 total = 0
14
15
16 def plot(x: int, y: int):
17     global total
18     total += 1
19     plt.plot(1, y, label=data.columns[y])
20     plt.scatter(data.iloc[:, x], data.iloc[:, y])
21     plt.xlabel(data.columns[x], fontproperties=font)
22     plt.ylabel(data.columns[y], fontproperties=font)
23     plt.savefig(
24         path.join(
25             path.dirname(__file__),
26             f"../figures/comparison{str(total)}.eps",
27         )
28     )
29     plt.savefig(
30         path.join(
31             path.dirname(__file__),
32             f"../figures/comparison{str(total)}.png",
33         )
34     )
35     plt.close()
36
37 plot(1, 2)
38 plot(3, 4)
39
40 plot(1, 3)
41 plot(2, 4)

```

B 数据

表 3: 城乡居民人均可支配收入和支出

年份	城镇收入	城镇支出	农村收入	农村支出
1978	343	311	134	116
1979	387	361	161	135
1980	478	412	191	162
1981	458	457	223	191
1982	495	471	270	220
1983	526	506	310	248
1984	608	559	355	274
1985	739	673	398	317
1986	900	799	424	357
1987	1002	884	463	398
1988	1181	1104	545	477
1989	1376	1211	602	535
1990	1510	1279	686	585
1991	1701	1454	709	620
1992	2027	1672	784	659
1993	2577	2111	922	770
1994	3496	2851	1221	1017
1995	4283	3538	1578	1310
1996	4839	3919	1926	1572
1997	5160	4186	2090	1617
1998	5425	4322	2162	1590
1999	5854	4616	2210	1577
2000	6280	4993	2253	1670
2001	6860	5309	2366	1741
2002	7703	6030	2476	1834
2003	8472	6511	2622	1913
2004	9122	7182	2936	2185
2005	10493	7943	3255	2555
2006	11759	8697	3587	2829
2007	13786	9997	4140	3224
2008	15781	11243	4761	3661
2009	17175	12265	5153	3994
2010	19109	13472	5919	4382
2011	21810	15160	6977	5221

年份	城镇收入	城镇支出	农村收入	农村支出
2012	24565	16674	7917	5908
2013	26467	18488	9130	7185
2014	28844	19968	10189	8383
2015	31195	21392	11422	9223
2016	33616	23079	12363	10130
2017	36396	24445	13432	10955
2018	39251	26112	14617	12124

C 题目

题目：改革开放 40 年是我国经济飞速发展，经济总量连上新台阶的 40 年，也是全国人民共享改革发展成果，生活水平大幅提高的 40 年。2018 年，我国居民人均可支配收入达到了 28228 元，扣除价格因素，比 1978 年实际增长 24.2 倍，年均增长 8.5%。表 3 是国家统计局提供的 1978–2018 年间我国城镇和农村常住居民的人均可支配收入及消费支出情况的统计数据。某国际知名经济研究机构的高级研究人员预测，中国的这一发展趋势在未来还将进一步持续。

D 仓库

仓库：<https://github.com/7086cmd/essay-2023>