OurWork

Analysis based on data ‘SDA\_2020\_St\_Gallen.csv’

The confusion matrix is as follows:

IC1 IC2 IC3

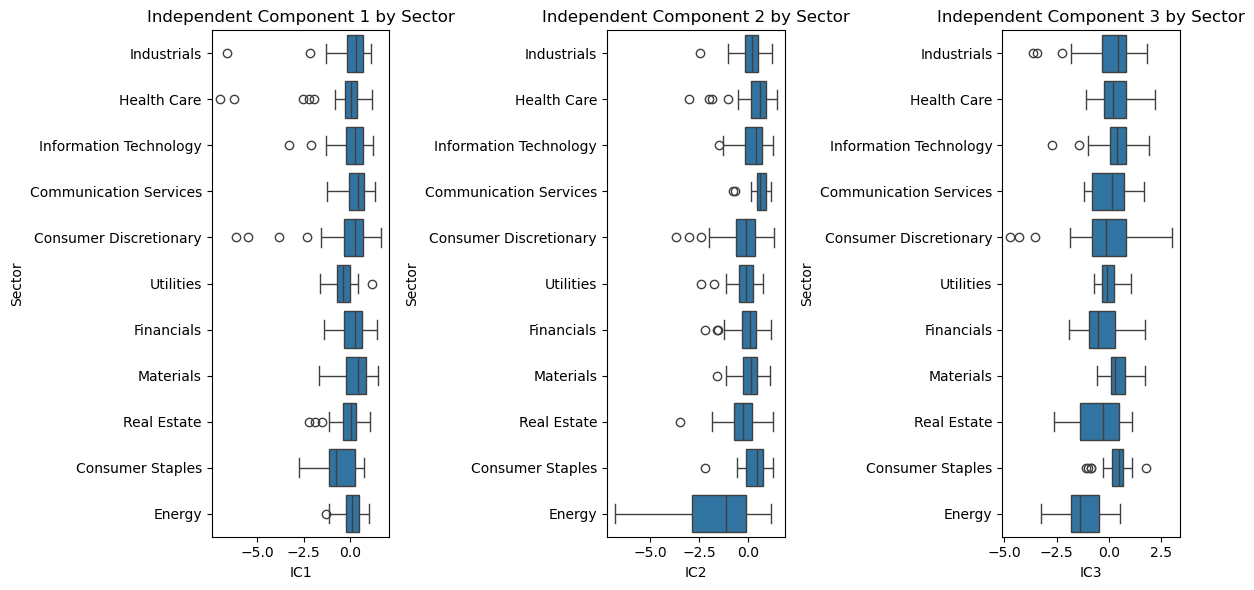
annual\_return\_log 0.030553 0.390005 0.800339

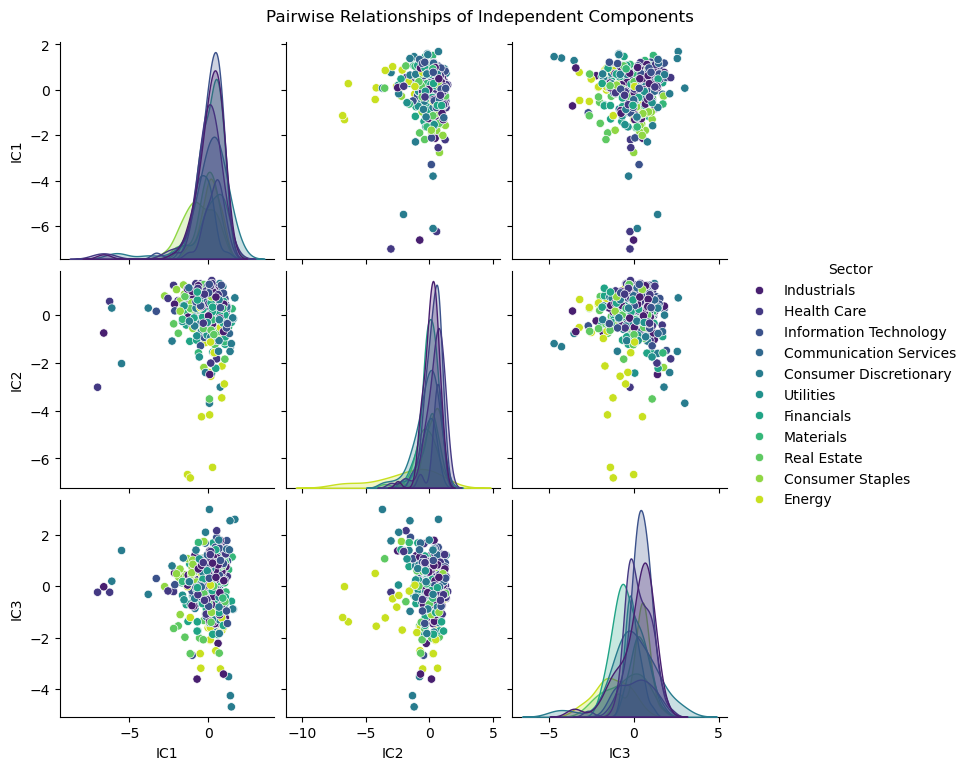
Std 0.255703 -0.495417 -0.650653

Skewness(偏度) -0.557034 0.780794 -0.215736

Kurtosis(峰度) -0.552071 -0.817559 0.143333

Annual\_return\_log:log(年收益率)





IC1 may represent “market systemic risk” and IC2 may represent “industry-specific risk patterns.”

From the plot we can see that the IC1 is very sensitive to Industrial indicator, Information indicator and Energy indicator.

IC2: Energy industry box has a low value, real estate and optional consumption is relatively dispersed(离散的), indicating that IC2 may capture the difference between the energy industry and other industries in a certain potential dimension (e.g. different drivers of the energy cycle and consumption / real estate).

IC3: Real Estate box size is wide, high discreteness, and the range of energy values is large.This reflect that IC3 describes the potential factors in the real estate and energy industries more complex, with significant differences in the industry.

In the future, we can add financial indicators such as liquidity, valuation indicators, etc. for a more comprehensive analysis of stock returns.

【Industry clustering: Points in the same industry have a certain aggregation in the multi-IC space, but there are also cross-sections (e.g. data points in the info technology and healthcare industries are partially overlapping). This illustrates:

Companies in the industry behave similarly (clustering) on the potential dimensions of independent component mapping, validating the association of ICs with industry characteristics;

There are cross-sections between industries, and I speculate that certain industries are driven by common market factors (e.g. technology + healthcare are both influenced by innovation and policy).】

IN CONCLUSION, these independent components likely represent the latent factors driving stock returns:

IC1 may be related to market cycles/fundamental asset volatility (outliers in energy and consumer staples correspond to cyclical industries);

IC2 may be associated with economic growth/consumer-driven factors;

IC3 may reflect risk appetite/asset allocation(the dispersion in real estate and energy corresponds to high-volatility/cyclical sectors).

In the future, companies can pay more attention on the combination of the information technology field and the healthcare field, boosting the increment of the annual stock return.

"""DEDA Unit 4OOP and Web Scraping FrameworkAuthors: Isabell Fetzer and Junjie Hu"""

"""Web Scraping encompasses any method allowing for extracting data from websites. Requests allows us to send an HTTP request to a webpage. BeautifulSoup parses the HTML in order to retrieve the desired information. Project: We wish to scrape the first page of the South China Morning Post’s news website. We acquire data about the news title, the news link and the news publication date and produce a tabular output stored as .csv file. """*# Load modules***import** requests**from** bs4 **import** BeautifulSoup **as** soup**from** datetime **import** datetime, date

*# Receiving source code from the South China Morning Post website*scmp\_url **=** 'https://www.scmp.com/knowledge/topics/china-economy/news'url\_request **=** requests**.**get(scmp\_url)

*# Returns the content of the response*url\_content **=** url\_request**.**content

*# Using BeautifulSoup to parse webpage source code*parsed\_content **=** soup(url\_content, 'html.parser')

*# Find all news sections*filtered\_parts **=** parsed\_content**.**find\_all('div', class\_**=**"sc-1yocfo6-0")page\_info **=** []

*# For loop iterates over every line in text***for** section **in** filtered\_parts:

unit\_info **=** {}

*# (1) Filter title, link, and text content*

filtered\_part1 **=** section**.**find\_all('a', class\_**=**"sc-1ij6sn6-0")

**if** len(filtered\_part1) **<** 2:

**continue**

*# Extract the title and link from the section*

news\_title **=** filtered\_part1[1]**.**text**.**strip() **if** len(filtered\_part1) **>** 1 **else** ''

news\_link **=** filtered\_part1[1]**.**get('href')**.**strip() **if** len(filtered\_part1) **>** 1 **else** ''

news\_link **=** f"https://www.scmp.com{news\_link}" *# adjust the relative link*

*# Filter the description text (optional if needed)*

news\_text **=** filtered\_part1[0]**.**text**.**strip() **if** len(filtered\_part1) **>** 0 **else** ''

*# (2) Filter date*

filtered\_part2 **=** section**.**find\_all('time', datetime**=True**)

**if** filtered\_part2:

**try**:

*# Parse the date format (example format: 2 Aug 2024 - 10:15PM)*

news\_date **=** datetime**.**strptime(filtered\_part2[0]**.**text**.**strip(), '%d %b %Y - %I:%M%p')

news\_date **=** news\_date**.**date() *# only keep the date part*

**except** ValueError:

*# If parsing fails, fallback to today's date*

news\_date **=** date**.**today()

**else**:

news\_date **=** date**.**today()

*# Add all info into the dictionary*

unit\_info['news\_title'] **=** news\_title

unit\_info['news\_link'] **=** news\_link

unit\_info['news\_text'] **=** news\_text

unit\_info['news\_date'] **=** news\_date

page\_info**.**append(unit\_info)

*# Print the collected information***for** info **in** page\_info:

print(info)