

《大学计算机程序设计（Python）》

实 践 作 业



作品名称：北京上海与哈尔滨空气质量分析

填写日期：2026年10月20日

目 录

第一章 背景介绍.....	1
第二章 作品简介.....	1
第三章 作品实现.....	2
第四章 制作经验总结.....	23
第五章 作品源代码与 AI 提示词截图.....	24
参考文献.....	49

预览版仅供参考请以最终提交版为准

第一章 背景介绍

本作品比较北京、上海与哈尔滨 2010—2024 年的空气质量历史记录。PM2.5 可核验序列覆盖 2013—2024 年，按公布值计算，三市期末较期初分别下降 65.7%、54.8%和 50.6%。2010—2012 年以 PM10、SO2、NO2 和旧口径达标天数补充历史背景，不编造早期 PM2.5 值。

北京和上海分别位于京津冀、长三角，哈尔滨补充了东北城市案例。三市仅代表自身，不能代替整个城市群，也不能据此评价全国。报告包括 2010—2019 年历史数据和 2020—2024 年近期数据，固定截至 2024 年，不声称反映最新年度状况。

主要问题是：较长时期的污染物浓度如何变化；近五年不同污染物变化方向是否一致；PM2.5 下降时优良天数占比是否必然上升。旧标准达标率与新标准 AQI 优良率分开保留，不跨口径计算改善幅度。优良天数占比是年度汇总指标，不是逐日 AQI 数列或年均 AQI。

第二章 作品简介

作品按“官方资料下载—核录—Pandas 清洗—统计分析—Matplotlib 绘图—报告解读”组织。年度主表共 45 条记录，其中 30 条来自 2010—2019 年历史表；另有 6 条 2020/2024 年污染物记录和 2 条污染日构成记录。来源索引含 29 项官方资料，历史表与年度主表重叠，不重复计数。本轮再加入 55 条气态污染物、等级天数、季节与环境专题条目；不同表存在城市年度重叠，不能相加当作独立样本。

Pandas 用于数据清洗，NumPy 用于数值核验，Matplotlib 与 WordCloud 生成 10 种图形。报告共 11 个图号，其中图 10 分 A、B 两页，共 12 张图。六个程序依次完成下载、核验、常规绘图、词频统计、矩阵瀑布与词云、季节哑铃图；原件快照支持离线复现。

小组组成与分工

小组共 6 名成员，任务分工及对应成果如下表所示。

学号	主要任务	交付内容
8	选题设计、进度协调；数据采集与口径审核；程序开发、统计分析、图表设计及报告整合	总体方案、完整程序、分析结论、十类图表及最终报告
12	资料链接与文件整理	资料目录
9	表格录入及单位核对	表格核对记录
13	图表标题与图例检查	图表文字检查记录
14	参考文献格式整理	参考文献格式清单
11	页码、文件命名及归档检查	提交文件清单

第三章 作品实现

3.1 数据来源与原始数据

2020—2024 年北京、上海资料来自两市生态环境局[1—10]，哈尔滨来自年度统计公报[11—15]及省级报告[16]。2010—2019 年采用国家统计局环境统计资料及《中国统计年鉴》[17—26]。程序下载网页、PDF 或年鉴图表并记录摘要，再按原表位置人工核录 CSV，不将人工录入描述成自动解析。

表 1 保留 2020—2024 年的 15 条年度记录，新增的 2010—2019 年 30 条记录见后续历史表。PM2.5 单位为 $\mu\text{g}/\text{m}^3$ ，优良天数占比单位为%。京沪保留各年度公报公布比例；哈尔滨 2020 年 82.8% 由 303 个优良日除以 366 个日历日折算，不冒充公报直接公布比例。

城市	年份	PM2.5	优良天数占比	来源
北京	2020	38	75.4	B20
北京	2021	33	78.9	B21
北京	2022	30	78.4	B22
北京	2023	32	74.2	B23
北京	2024	30.5	79.2	B24
上海	2020	32	87.2	S20
上海	2021	27	91.8	S21
上海	2022	25	87.1	S22
上海	2023	28	87.7	S23
上海	2024	28	88.5	S24
哈尔滨	2020	47	82.8*	H20
哈尔滨	2021	37	84.9	H21
哈尔滨	2022	38	84.9	H22
哈尔滨	2023	36	83.3	H23
哈尔滨	2024	40	85.2	H24

表 1 B20—B24 对应[1—5]，S20—S24 对应[6—10]，H20—H24 对应[11—15]。*为按日历年折算值。哈尔滨 2024 年分母不一致问题见下页。

原始数据与统计口径 续

表 2 为跨污染物比较的全部起止年度数据。四种污染物均采用年均浓度，单位统一为 $\mu\text{g}/\text{m}^3$ 。PM2.5 是 PM10 中粒径更小的一部分，不能把两者直接相加后解释为总颗粒物浓度。

城市	年份	PM2.5	PM10	NO2	SO2	来源
北京	2020	38	56	29	4	B20
北京	2024	30.5	54	24	3	B24
上海	2020	32	41	37	6	S20
上海	2024	28	43	30	7	S24
哈尔滨	2020	47	未列示	32	17	H20
哈尔滨	2024	40	62	29	10	H24/P24

哈尔滨 2024 年 PM10 为 $62 \mu\text{g}/\text{m}^3$ ，取省生态环境厅报告印刷第 19 页 [16]，pm10_source_id 单独关联 P24；其余三项取市统计公报 [15]。2020 年 PM10 未取得可核验值，保留缺失，所以图 3 相应变化率为灰格。缺失不等于 0，不参与计算。

哈尔滨 2024 年公报同时列出有效监测 365 天、优良 312 天和 85.2% [15]。 $312 \div 365 \approx 85.5\%$ ， $312 \div 366 \approx 85.2\%$ ，三项无法按“有效监测日为分母”同时成立。报告保留 85.2% 这个公布值，并将跨城优良率比较和相关系数限定为探索性结果，不自行纠正原文。

哈尔滨 2022 年 PM2.5 采用该年统计公报的 $38 \mu\text{g}/\text{m}^3$ [13]。不同发布材料中的同期值可能存在差异，本文固定使用年度统计公报序列，不根据后续新闻中的同比倒推年值。原始公报注明部分数据为初步统计，正式年鉴修订值可能不同。

图 6 的补充数据来自上海 2022 年公报：47 个污染日中，O3 为首要污染物的有 41 天，PM2.5 为首要污染物的有 6 天 [8]。这两个类别在该公报的该年度统计中合计为 47 天，仅用于描述这些污染日的构成。

历史数据补充：北京（2010—2019 年）

下表浓度均为 $\mu\text{g}/\text{m}^3$ ，天数为“空气质量达到及好于二级的天数”。2010—2012 年浓度原单位为 mg/m^3 ，已乘 1000 换算；旧达标率为原表公布值。2013 年起新增 PM2.5 等评价指标，前后达标天数和比例不直接比较[27]。

年份	PM2.5	PM10	SO2	NO2	达标天数	旧达标率 (%)
2010	—	121	32	57	286	78.4
2011	—	113	28	56	286	78.4
2012	—	109	29	52	281	76.8
2013	89	108	26	56	167	—
2014	86	116	22	57	168	—
2015	81	102	14	50	186	—
2016	73	92	10	48	198	—
2017	58	84	8	46	226	—
2018	51	78	6	42	227	—
2019	42	68	4	37	240	—

来源：国家统计局 [17—26]，分别对应 2010—2019 年；CSV 中为 N10—N19。“—”表示原表未列该指标。北京的历史 PM2.5 值按原表整数精度保留，不从同比或其他资料反推。

CO 与 O3 补充记录（2013—2019 年）

年份	CO 日均 P95 (mg/m^3)	O3 日最大 8 小时 P90 ($\mu\text{g}/\text{m}^3$)
2013	3.4	188
2014	3.2	200
2015	3.6	203
2016	3.2	199
2017	2.1	193
2018	1.7	192
2019	1.4	191

CO 和 O3 为不同时间尺度的百分位浓度，不能当作年平均值与上表直接相加或混用。历史达标天数保持原值，不假定有效监测日等于日历天数，因此不据此生成新口径优良率。

历史数据补充：上海（2010—2019 年）

下表浓度均为 $\mu\text{g}/\text{m}^3$ ，天数为“空气质量达到及好于二级的天数”。2010—2012 年浓度原单位为 mg/m^3 ，已乘 1000 换算；旧达标率为原表公布值。2013 年起新增 PM2.5 等评价指标，前后达标天数和比例不直接比较[27]。

年份	PM2.5	PM10	SO2	NO2	达标天数	旧达标率 (%)
2010	—	79	29	50	336	92.1
2011	—	80	29	51	337	92.3
2012	—	71	23	46	343	93.7
2013	62	84	24	48	246	—
2014	52	71	18	45	278	—
2015	53	69	17	46	252	—
2016	45	59	15	43	276	—
2017	39	55	12	44	275	—
2018	36	51	10	42	295	—
2019	35	45	7	42	309	—

来源：国家统计局 [17—26]，分别对应 2010—2019 年；CSV 中为 N10—N19。“—”表示原表未列该指标。上海的历史 PM2.5 值按原表整数精度保留，不从同比或其他资料反推。

CO 与 O3 补充记录（2013—2019 年）

年份	CO 日均 P95 (mg/m^3)	O3 日最大 8 小时 P90 ($\mu\text{g}/\text{m}^3$)
2013	1.6	158
2014	1.3	149
2015	1.5	161
2016	1.3	164
2017	1.2	181
2018	1.1	160
2019	1.1	151

CO 和 O3 为不同时间尺度的百分位浓度，不能当作年平均值与上表直接相加或混用。历史达标天数保持原值，不假定有效监测日等于日历天数，因此不据此生成新口径优良率。

历史数据补充：哈尔滨（2010—2019 年）

下表浓度均为 $\mu\text{g}/\text{m}^3$ ，天数为“空气质量达到及好于二级的天数”。2010—2012 年浓度原单位为 mg/m^3 ，已乘 1000 换算；旧达标率为原表公布值。2013 年起新增 PM2.5 等评价指标，前后达标天数和比例不直接比较[27]。

年份	PM2.5	PM10	SO2	NO2	达标天数	旧达标率 (%)
2010	—	101	45	48	313	85.8
2011	—	99	41	46	317	86.8
2012	—	94	36	47	319	87.2
2013	81	119	44	56	239	—
2014	72	111	57	52	241	—
2015	70	103	40	51	227	—
2016	52	74	29	44	282	—
2017	58	84	25	44	271	—
2018	39	65	20	37	310	—
2019	42	67	17	32	304	—

来源：国家统计局 [17—26]，分别对应 2010—2019 年；CSV 中为 N10—N19。“—”表示原表未列该指标。哈尔滨的历史 PM2.5 值按原表整数精度保留，不从同比或其他资料反推。

CO 与 O3 补充记录（2013—2019 年）

年份	CO 日均 P95 (mg/m^3)	O3 日最大 8 小时 P90 ($\mu\text{g}/\text{m}^3$)
2013	2.2	72
2014	1.6	111
2015	1.8	106
2016	1.8	103
2017	2	133
2018	1.3	136
2019	1.4	116

CO 和 O3 为不同时间尺度的百分位浓度，不能当作年平均值与上表直接相加或混用。历史达标天数保持原值，不假定有效监测日等于日历天数，因此不据此生成新口径优良率。

3.2 数据清洗与统计方法

统一城市名称与数值类型，以“城市+年份”为年度表、历史表和端点表主键，以“城市+年份+污染物”为构成表主键。必需字段遇到缺失、重复、非数值、无穷值或负值即报错。早期 PM2.5 等已声明缺失项保留 NA，不填零、不插值，不自动前向填充。

核验覆盖三市 2010—2024 年，共 45 条年度记录；其中 2010—2012 年 PM2.5 缺失 9 条，2013—2024 年有效 36 条。历史表 30 条与年度表按主键及 PM2.5 核对。旧口径比例与历史达标天数不并入 AQI 优良率列，该列早期 30 条留空。端点表 PM10 缺失 1 项，重复主键 0 项，插补 0 项。

口径核对中的两个问题

第一，上海 2024 年公报 PDF 第 5 页公布优良率 88.5%，优 132 天、良 191 天，六类等级天数合计 365 天[10]。 $323 \div 365$ 约为 88.5%，而用 366 个日历日作分母约为 88.3%。原件未在此处写明“扣沙尘前后”的区别，因此采用 88.5% 公布值，不擅自解释相差的一天。监测点调整、有效天数与评价口径均限制跨期绝对比较。

第二，北京 2023 年公开 PM2.5 年值为 32，2024 年为 30.5[4—5]。按这两个公布值计算同比约为 -4.7%，而 2024 年发布稿给出的官方同比降幅为 6.2%。这提示公布精度或评价口径可能存在差异；未取得未取整底表，不能自行认定原因或倒推修改 2023 年数据。本作品的同比列仅标记为“基于公布值计算”，不替代官方同比。

计算方法

浓度变化率 = $(\text{期末浓度} - \text{期初浓度}) \div \text{期初浓度} \times 100\%$ 。负值表示下降。为便于文字解读，累计降幅另按 $(\text{期初浓度} - \text{期末浓度}) \div \text{期初浓度} \times 100\%$ 计算，下降时为正值。

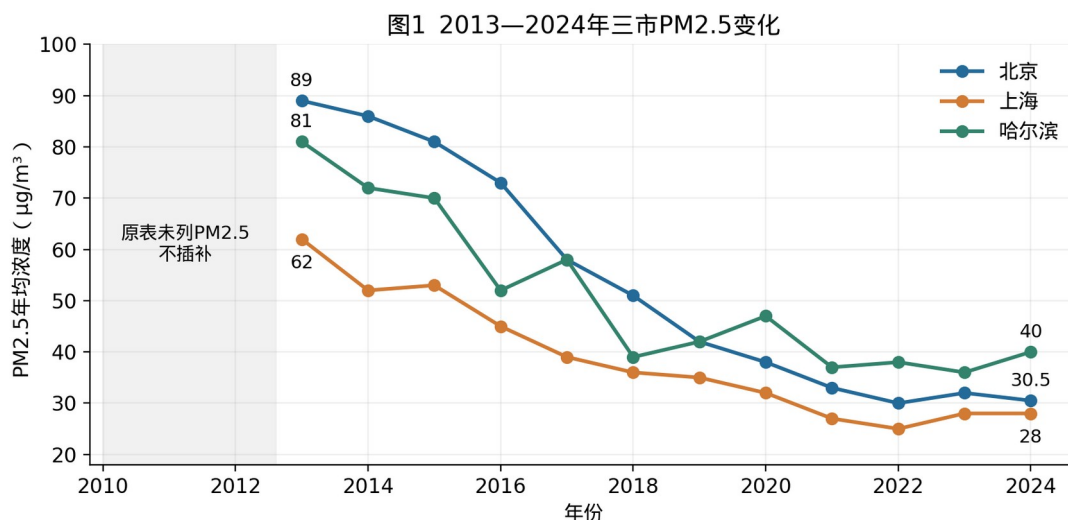
优良率变化 = 期末优良率 - 期初优良率，单位为“个百分点”。例如北京 $79.2\% - 75.4\% = 3.8$ 个百分点，不能直接写成“提高 3.8%”。

PM2.5 的均值、中位数、极差使用 2013—2024 年每市 12 个年度值。Pearson 相关系数仍仅按 2020—2024 年每市 5 个点计算，避免把旧标准达标率或未经确认分母的历史天数换算率并入同一序列。哈尔滨比例口径限制仍然保留；相关系数不作为显著性或因果证据。

复现方式

可选联网执行 00_download.py（需 curl）；离线依次执行 01_prepare.py、02_visualize.py、03_corpus.py、04_advanced.py、05_seasonal.py。先安装 requirements.txt 依赖并配置中文字体。下载失败保留日志；核录数据与包内官方快照支持离线复现。

3.3 长期 PM2.5 总体改善但存在反复



来源：国家统计局[20]—[26]、三市官方公报[1]—[15]；2010—2012年原表未列PM2.5。

图1表明，2013—2024年三市PM2.5年均浓度总体下降。北京由89降至30.5 $\mu\text{g}/\text{m}^3$ ，累计下降65.7%；上海由62降至28 $\mu\text{g}/\text{m}^3$ ，下降54.8%；哈尔滨由81降至40 $\mu\text{g}/\text{m}^3$ ，下降50.6%。北京的绝对降幅和相对降幅均为三市最高，说明不同城市的长期变化幅度存在差异。2010—2012年原始资料未列示PM2.5，因此不纳入该指标的趋势分析。

三市长序列均呈总体下降，但有不同转折。北京2013—2016年依次为89、86、81、73，2017年降至58；上海2014年的52在2015年回升至53；哈尔滨2016年的52在2017年回升至58。长期端点下降不能解释为每一年都在改善。

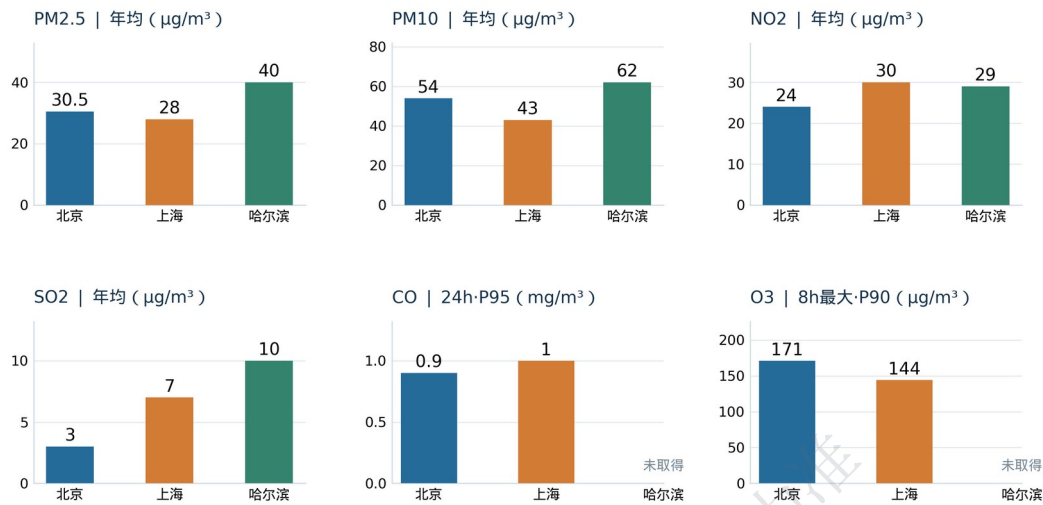
长期序列采用各来源公布的浓度值及精度。2019年前的数据主要来自国家统计局，2020年起主要采用地方年度公报；监测网络、有效样本及发布口径的差异可能影响跨期可比性。因此，本节分析用于描述公开统计序列的变化，不直接识别单项治理措施的效果。

从2020—2024年的近期变化看，北京、上海和哈尔滨PM2.5分别下降19.7%、12.5%和14.9%。与此同时，北京、上海2023年的浓度均较2022年回升，哈尔滨2024年较2023年上升11.1%。上述结果表明，长期下降趋势与短期年际波动并存，空气质量评价需要同时关注累计变化和近期表现。

3.4 不同污染物呈现不同城市差异

图2 2024年六项污染物比较

同项比较城市；各面板独立刻度，不能跨面板比较柱高。



CO与O3为百分位浓度；哈尔滨两项缺值不补零。

来源：2024年官方公报[5][10][15][16][28]。

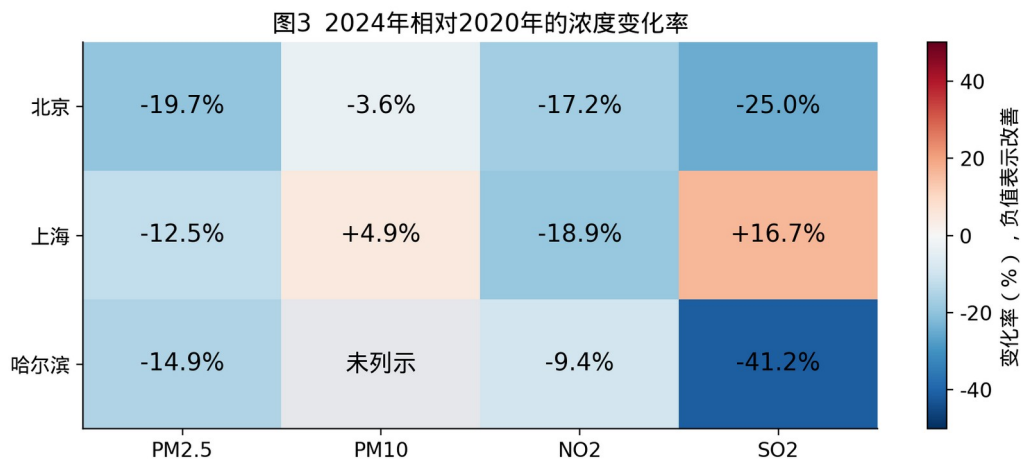
图2比较了2024年三市六项空气污染物的浓度水平。PM2.5、PM10、NO2和SO2采用年均浓度，CO采用24小时平均第95百分位浓度，O3采用日最大8小时平均第90百分位浓度。其中CO的单位为mg/m³，其余指标为μg/m³。各面板采用独立刻度，用于比较同一污染物的城市差异。

哈尔滨PM2.5、PM10、NO2和SO2分别为40、62、29和10 μg/m³[15—16]，其中两项颗粒物及SO2均高于北京、上海。北京和上海CO分别为0.9、1.0 mg/m³，O3分别为171、144 μg/m³[10][28]，两项指标的相对高低顺序不同。哈尔滨CO和O3因缺少可核验值，未纳入相应比较。

北京与上海的PM10、NO2、PM2.5和SO2浓度差分别为11、6、2.5和4 μg/m³。其中，北京颗粒物浓度高于上海，NO2和SO2浓度则低于上海。这说明城市间的空气质量差异具有污染物特异性，单一指标不足以概括整体状况。

综合六项指标，三市的污染物浓度结构存在明显差异，监测和分析应兼顾颗粒物与气态污染物。由于各指标的评价限值、形成过程及统计口径不同，本研究不以数值大小进行跨污染物危害排序；污染源及影响因素仍需结合排放清单、气象条件和站点监测资料进一步分析。

3.5 统一基期后识别改善幅度



来源：京沪生态环境局、哈尔滨市统计局及黑龙江省生态环境厅[1]—[16]。
起止年变化率由程序计算；灰格表示缺少可核验PM10年值。

图3以2020年为基期，比较2024年四项年均浓度的变化率。北京PM2.5、PM10、NO2和SO2分别下降19.7%、3.6%、17.2%和25.0%，四项指标均有所改善。其中PM10降幅相对较小，反映出不同污染物的变化速度并不一致。

上海PM2.5和NO2分别下降12.5%和18.9%，PM10和SO2则分别上升4.9%和16.7%。哈尔滨PM2.5、NO2和SO2分别下降14.9%、9.4%和41.2%，其中SO2降幅最大。哈尔滨2020年PM10缺少可核验数据，未计算该项变化率。

相对变化率需要结合基期浓度解释。上海SO2由6增至7 $\mu\text{g}/\text{m}^3$ ，绝对增量为1 $\mu\text{g}/\text{m}^3$ ，相对增幅为16.7%；北京SO2由4降至3 $\mu\text{g}/\text{m}^3$ ，绝对减量同为1 $\mu\text{g}/\text{m}^3$ ，相对降幅为25.0%。两市相同规模的绝对变化对应不同百分比，说明低基数会影响相对变化率的大小。

总体来看，三市在多数已知指标上呈现下降趋势，但改善幅度及方向存在差异。因此，空气质量分析应同时报告原始浓度与变化率，并结合图1考察年际过程。图3反映两个端点之间的净变化，不代表期间各年度均沿同一方向变动。

3.6 PM2.5 与优良率不是一一对应

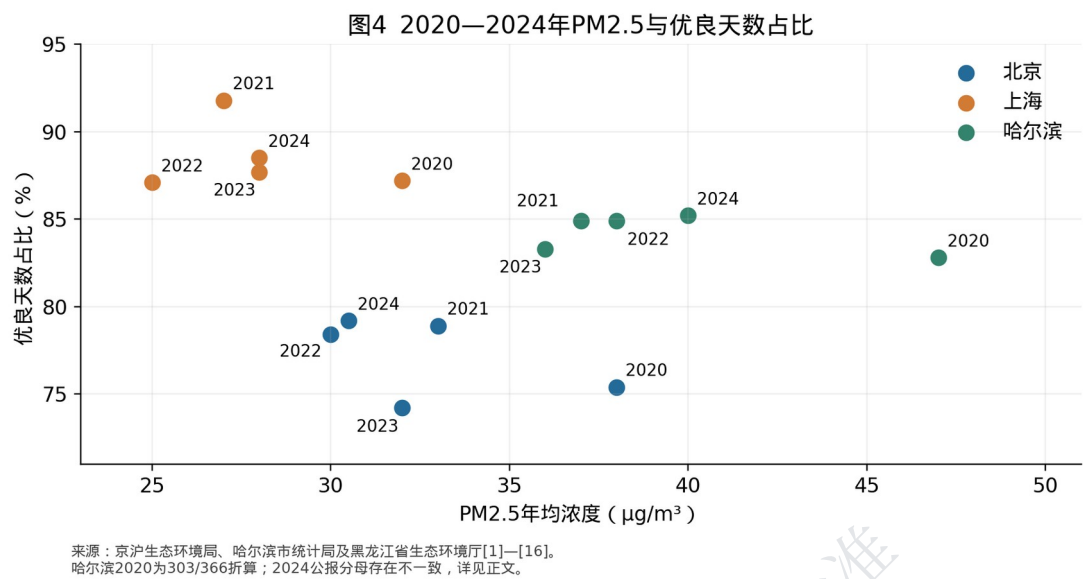


图4展示2020—2024年三市PM2.5年均浓度与空气质量优良天数占比的关系，每个点对应一个城市年度。研究期内，北京优良率由75.4%升至79.2%，上海由87.2%升至88.5%，分别提高3.8和1.3个百分点。散点仅采用这一时期的记录，以减少早期评价标准差异对比较的影响。

按城市计算，PM2.5与优良率的Pearson相关系数分别为北京-0.503、上海-0.217、哈尔滨-0.505，均呈负相关。其中上海的线性关联较弱。由于每市仅有5个年度样本，且哈尔滨部分比例的分母口径不一致，这些结果仅作为探索性描述，不用于预测或因果判断。

上海2021—2022年的变化体现了两项指标的区别：PM2.5由27降至25 μg/m³，而优良率由91.8%降至87.1%，下降4.7个百分点。PM2.5年均浓度反映单一污染物的年度平均水平，优良率则取决于综合空气质量评价结果。因此，PM2.5下降并不必然伴随优良率同步提高。

哈尔滨2020—2024年的优良天数占比依次为82.8%、84.9%、84.9%、83.3%和85.2%。其中2020年为日历年折算值，2024年公报公布比例与所列天数存在分母不一致问题，首末2.4个百分点的差值应谨慎解释。后续研究宜在统一有效监测日口径的基础上，结合其他污染物共同分析优良率变化。

3.7 描述十二年年度值的集中与离散

图5 2013—2024年PM2.5年度值分布

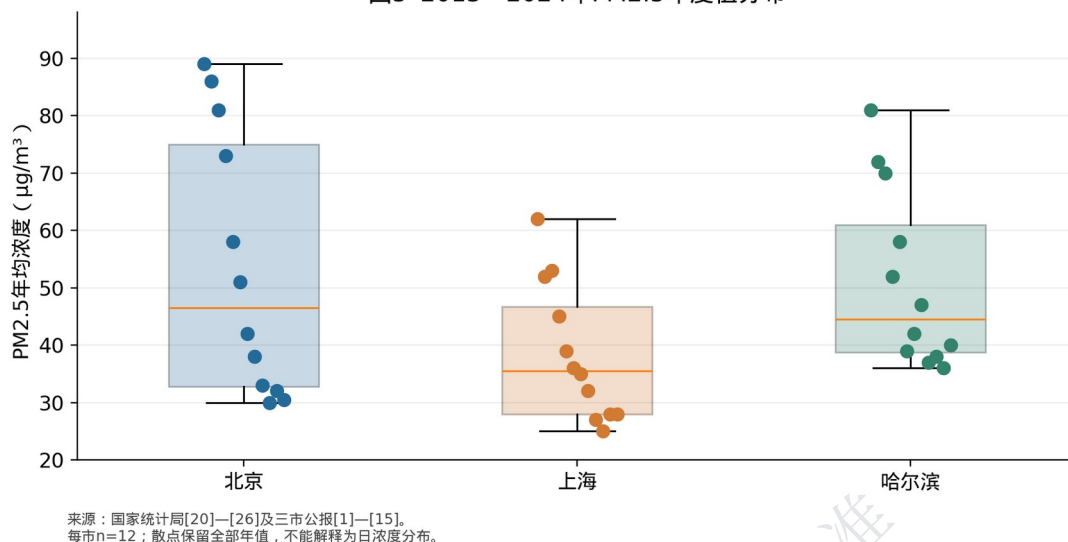


图5概括了2013—2024年三市PM2.5年均浓度的分布特征，每市包含12个年度观测值。北京均值为 $53.625\text{ }\mu\text{g}/\text{m}^3$ ，中位数 $46.5\text{ }\mu\text{g}/\text{m}^3$ ，取值范围为 $30\text{—}89\text{ }\mu\text{g}/\text{m}^3$ ；上海均值为 $38.5\text{ }\mu\text{g}/\text{m}^3$ ，中位数 $35.5\text{ }\mu\text{g}/\text{m}^3$ ，取值范围为 $25\text{—}62\text{ }\mu\text{g}/\text{m}^3$ 。上海在该研究时段内的均值、中位数和极差均低于北京。

哈尔滨均值为 $51\text{ }\mu\text{g}/\text{m}^3$ ，中位数 $44.5\text{ }\mu\text{g}/\text{m}^3$ ，取值范围为 $36\text{—}81\text{ }\mu\text{g}/\text{m}^3$ ，极差为 $45\text{ }\mu\text{g}/\text{m}^3$ 。三市极差依次为北京59、哈尔滨45、上海 $37\text{ }\mu\text{g}/\text{m}^3$ ，显示年度浓度分布的离散程度不同。图中箱体表示中间50%的年度值，须线采用1.5倍四分位距规则，并叠加全部年度观测点。

上述分布同时包含长期下降和年际波动的影响。北京早期浓度较高、后期下降幅度较大，因此较宽的分布范围不宜单独解释为空气质量不稳定。结合图1可以进一步区分时间趋势与年度反复，避免忽略观测值的先后顺序。

箱线图的统计对象为年度平均浓度，其结果不能直接代表逐日波动、单个监测站状况或个人暴露水平。2010—2012年缺失的PM2.5值未参与计算。若进一步研究季节差异及污染过程持续时间，需要使用日尺度或更高频率的监测数据。

3.8 污染日构成与空气质量等级

图6 上海污染日构成与空气质量等级

左图分母为污染日；右图分母为公报列示的全年等级合计天数。

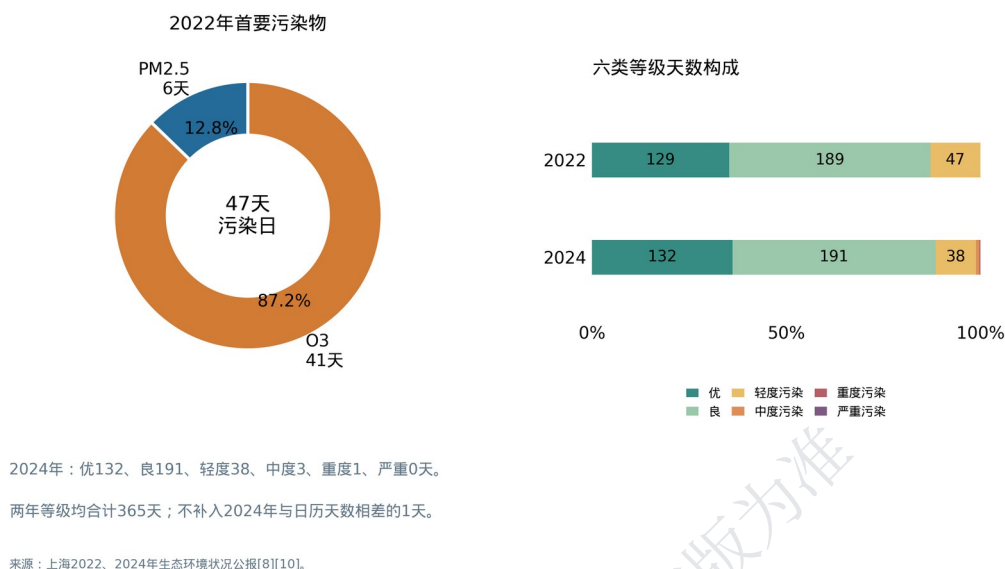


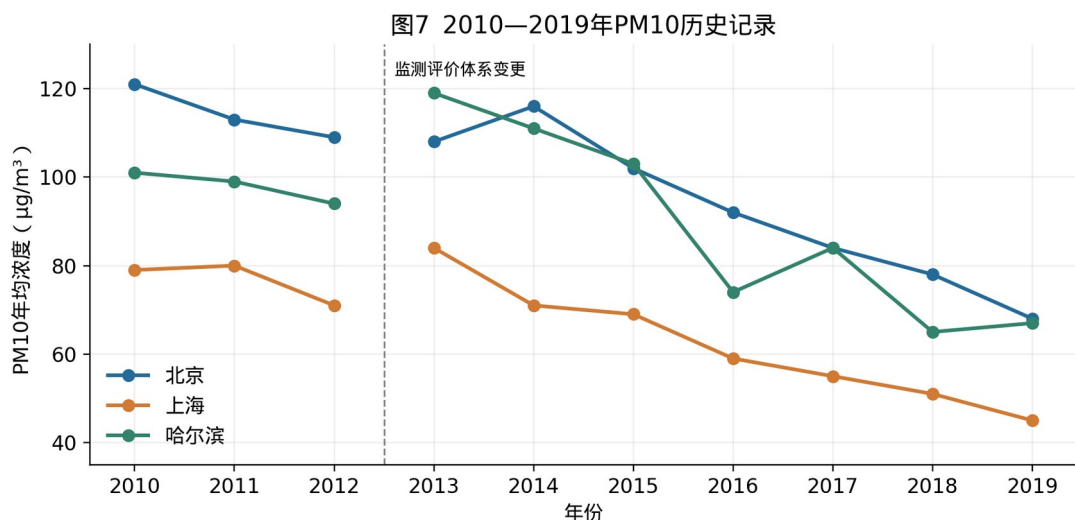
图6从首要污染物和空气质量等级两个角度分析上海的污染状况。2022年47个污染日中，以O3为首要污染物的有41天，占87.2%；以PM2.5为首要污染物的有6天，占12.8%[8]。臭氧在该年度污染日构成中占比较高，表明仅关注颗粒物不足以全面认识空气质量状况。

这一构成与图4中“PM2.5下降而优良率未同步提高”的现象相呼应，说明两项评价指标反映的信息不同。但首要污染物构成仅描述污染日的类别分布，不能据此精确分解各污染物对年度优良率变化的贡献，也不等同于污染物质量占比。

从空气质量等级看，上海优良天数由2022年的318天增加至2024年的323天；2024年仍有3天中度污染和1天重度污染[8][10]。这表明年度总体表现改善与部分较重污染过程可以同时出现。两年等级构成均以公报列示的365天为分母，与左侧首要污染物构成所用的47个污染日口径不同。

综合来看，空气质量评价需要同时关注优良天数、污染程度与首要污染物类型。颗粒物和臭氧应纳入共同的分析框架，并结合污染过程的发生时段及持续时间，进一步识别不同指标之间的变化关系。

3.9 早期 PM10 记录与评价口径边界



来源：国家统计局[17]—[26]；早期mg/m³乘1000统一单位。
2012/2013处不连线；浓度历史展示不等于新旧空气质量评价结果可直接相比。

图7展示2010—2019年三市PM10浓度的历史变化。2010—2012年，北京由121降至109 $\mu\text{g}/\text{m}^3$ ，上海由79降至71 $\mu\text{g}/\text{m}^3$ ，哈尔滨由101降至94 $\mu\text{g}/\text{m}^3$ ，三市期末浓度均低于期初。PM10记录为PM2.5尚未列示的早期阶段提供了颗粒物变化的参考。

在2013—2019年窗口中，北京PM10由108降至68，上海由84降至45，哈尔滨由119降至67 $\mu\text{g}/\text{m}^3$ ，按公布值计算分别下降37.0%、46.4%和43.7%。哈尔滨2017年PM10从上一年的74回升至84，与同期PM2.5由52升至58方向一致，但不能据此确定污染来源。

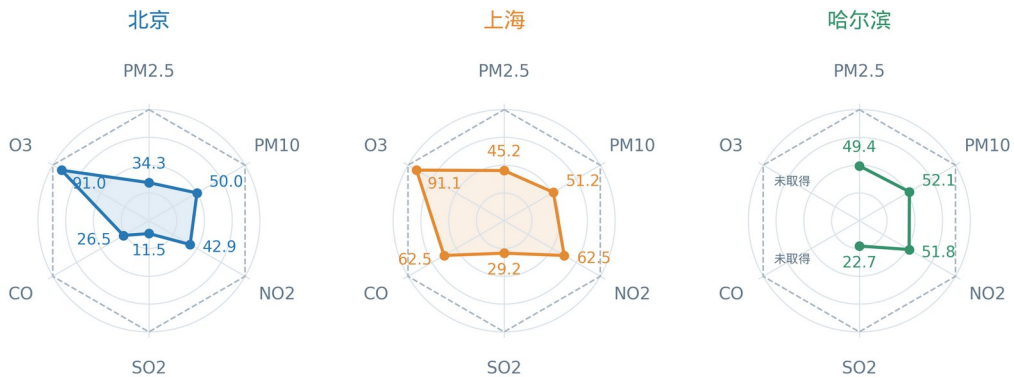
2013年前后监测评价体系发生变化，因此图7在2012年与2013年之间断开连线。虽然PM10浓度已统一单位，但监测覆盖与评价方式的变化仍可能影响跨期比较。早期旧标准达标率与后期新标准优良率不具有直接可比性，应分别讨论。

历史数据还包括SO₂、NO₂及2013年后的CO和O₃，为分析多污染物的长期变化提供了基础。跨期研究需要同时核对指标定义、单位、统计精度及来源范围。本报告将历史浓度趋势与近期空气质量比例分开分析，以保持统计口径的明确性。

3.10 六污染物雷达图与长期变化轮廓

图8 六项污染物的长期变化轮廓

2024年相对本市2013年的浓度指数 | 基期=100；数值越小，累计降幅越大



同心圆由内向外为25、50、75、100；虚线六边形为基期100。

CO采用24小时第95百分位；O3采用日最大8小时第90百分位。

哈尔滨两项缺值保留断线；不补零、不计算多边形综合得分。

来源：国家统计局[20]及2024年官方公报[5][10][15][16][28]；各项独立以2013=100。

图8以2013年为基期，对PM2.5、PM10、NO2、SO2、CO和O3六项指标进行标准化比较。各项2024年指数等于本市该项期末浓度与基期浓度之比乘以100，指数低于100表示较基期下降。三市采用相同半径刻度，以展示不同污染物的相对变化轮廓。

北京六项指数依次为34.3、50.0、42.9、11.5、26.5和91.0，上海依次为45.2、51.2、62.5、29.2、62.5和91.1。两市SO2的相对降幅均最大，O3降幅则相对较小；CO的累计降幅也明显大于O3。这表明不同污染物的长期变化存在显著差异，颗粒物及部分气态污染物的下降不能代表臭氧同步大幅改善。

哈尔滨PM2.5、PM10、NO2和SO2指数分别为49.4、52.1、51.8和22.7，四项指标均低于基期。CO与O3缺少可核验的2024年值，因此相应轴保留缺失。雷达图用于比较各指标相对变化，不以多边形面积构建综合得分；城市间的期末浓度比较仍需结合图2。

基期浓度取自国家统计局资料[20]，2024年浓度取自官方年度公报[5][10][15][16][28]。年均浓度与百分位浓度分别按自身口径计算指数，监测网络及来源变化仍是跨期比较的限制。完整基期值、期末值和指数见项目计算结果表。

3.11 云朵词云与扩展统计资料

图9 生态环境资料中的关键词云

13份官方资料节选 | 3,843个有效词次 | 真实词频前150项

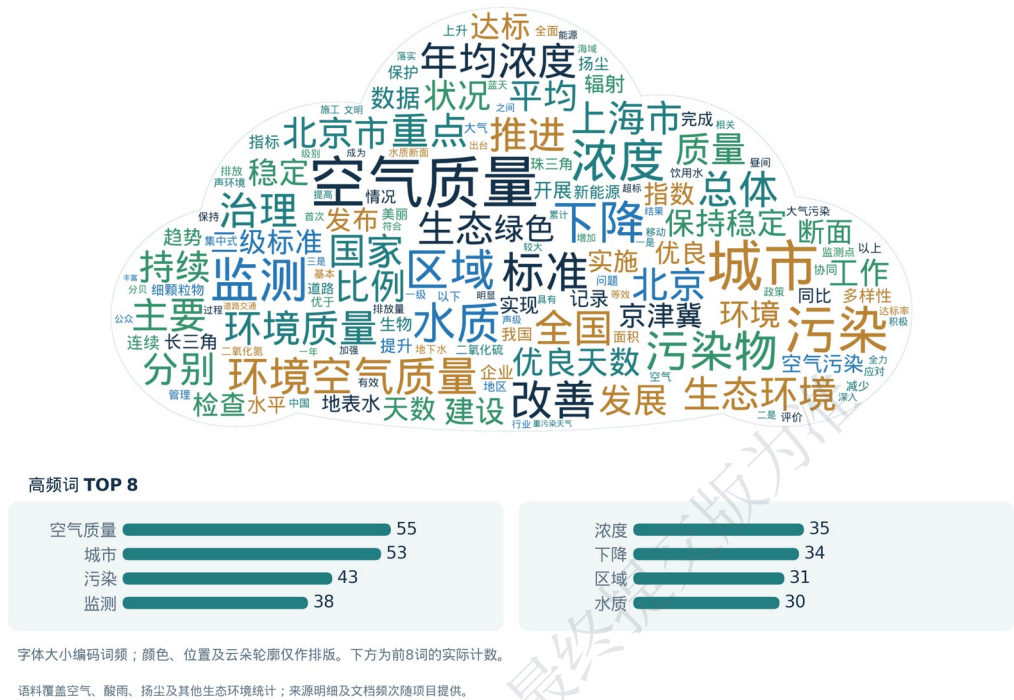


图9 基于13份官方生态环境资料节选开展词频分析，语料涉及空气质量、酸雨、扬尘、水质和声环境等内容。经分词、停用词过滤及精确重复段落去重，共得到3843个有效词次、1338个不同词项。词云展示频次最高的150个词项，下方统计条列示前8个高频词的实际计数。

“空气质量”“城市”“污染”的出现次数分别为55、53和43次，说明所选资料的表述主要围绕城市环境状况及污染问题展开。“监测”“浓度”“水质”等高频词则体现出语料兼具环境监测和多领域统计内容。词频结果用于概括所选文本的主题特征，不代表污染程度或各城市治理成效。

语料清单、统计口径与可复核性

来源	选取资料	有效词次
[2]	北京 2021 公报解读	715
[5]	北京 2024 空气质量报道	881
[27]	环境监测标准访谈	1131
[6]	上海 2020 公报发布稿	229
[7]	上海 2021 公报发布稿	202
[9]	上海 2023 质量概述	57
[8]	上海 2022 公报正文块	54
[10]	上海 2024 公报正文块	59
[13]	哈尔滨 2022 统计公报	40
[14]	哈尔滨 2023 统计公报	40
[15]	哈尔滨 2024 统计公报	43
[29]	全国 2024 公报发布稿	378
[16]	黑龙江 2024 采暖期段落	14

文本处理采用固定领域词典，将“生态环境”“空气质量”等识别为完整词项，并保留至少两个字的中文词。精确重复段落仅计一次，同义词及相近表述不作合并。原始节选、处理规则、摘要校验值与词频表均随项目保存，以支持结果复核。

“空气质量”占有效词次的 1.43%，出现在 7 份资料中；“监测”出现 38 次，覆盖 9 份资料。词频反映累计出现次数，文档频次反映涉及该词的资料数量，两者提供不同的文本观察角度。由于各资料篇幅及选题范围存在差异，本研究不依据词频对城市治理重点进行排名。

3.12 多污染物、分阶段矩阵式瀑布图

图10A 多污染物分阶段变化矩阵

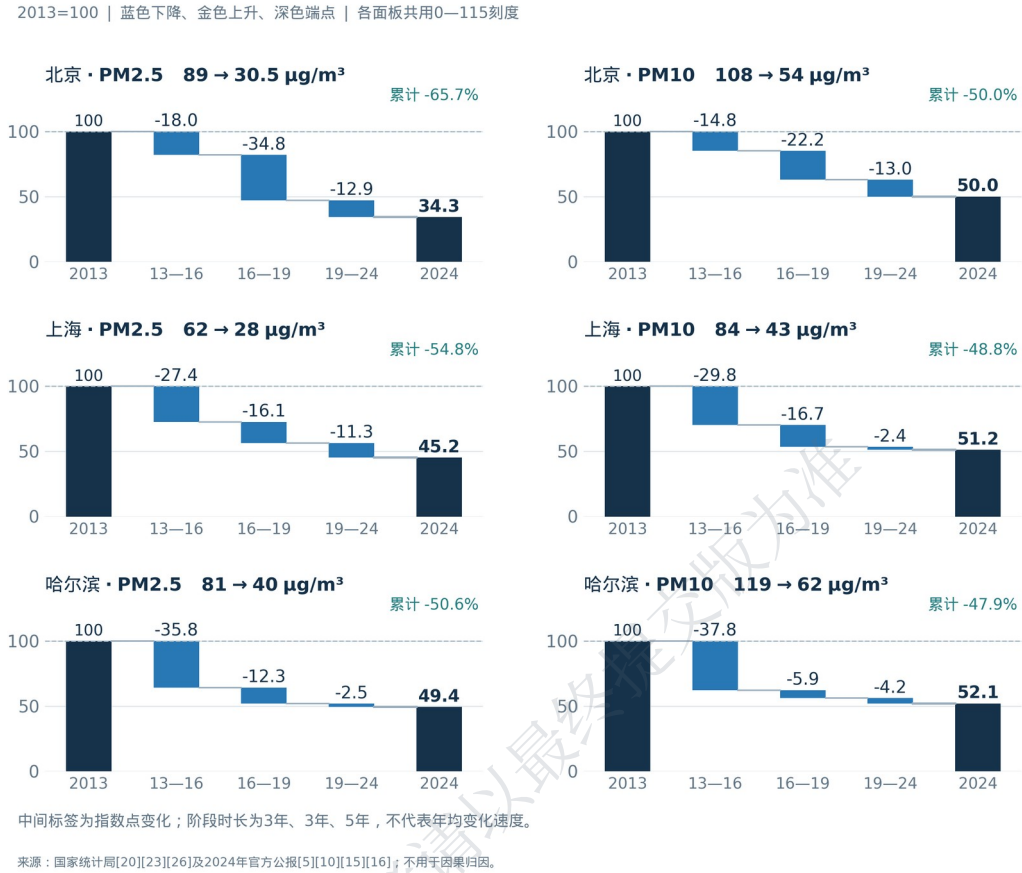


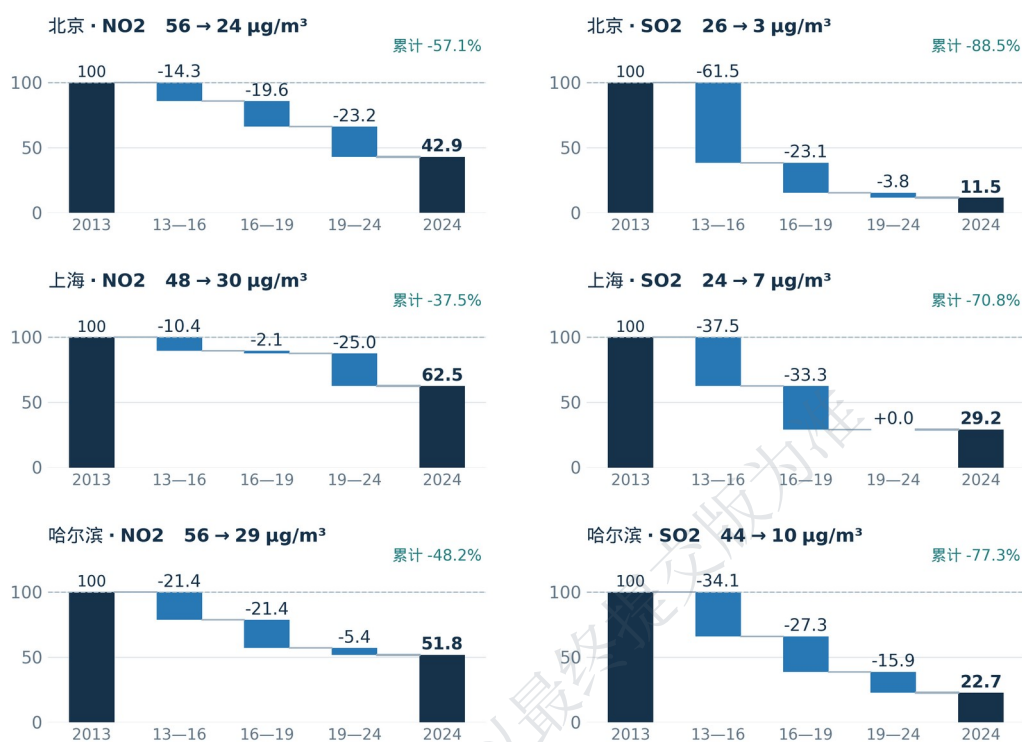
图 10A 和图 10B 采用分阶段矩阵瀑布图，比较三市四项污染物在 2013—2016 年、2016—2019 年和 2019—2024 年的变化。各市各项指标均以 2013 年浓度为 100，阶段增减表示指数点变化，三段变化与基期相加等于 2024 年指数。该方法将累计变化分解到具体时期，便于识别不同城市的主要下降阶段。

北京 PM2.5 三个阶段分别变化-18.0、-34.8 和-12.9 个指数点，最大降幅出现在 2016—2019 年；哈尔滨分别变化-35.8、-12.3 和-2.5 个指数点，最大降幅出现在 2013—2016 年。两市下降较快的阶段不同，反映出长期变化进程的差异。三个阶段分别持续 3、3 和 5 年，阶段累计降幅不等同于年均下降速度。

矩阵式瀑布图：气态污染物与解释边界

图10B 多污染物分阶段变化矩阵

2013=100 | 蓝色下降、金色上升、深色端点 | 各面板共用0—115刻度



中间标签为指数点变化；阶段时长为3年、3年、5年，不代表年均变化速度。

来源：国家统计局[20][23][26]及2024年官方公报[5][10][15][16]；不用于因果归因。

图 10B 显示，2013—2024 年三市 SO₂ 浓度均明显下降。北京由 26 降至 3 $\mu\text{g}/\text{m}^3$ ，累计下降 88.5%；上海由 24 降至 7 $\mu\text{g}/\text{m}^3$ ，下降 70.8%；哈尔滨由 44 降至 10 $\mu\text{g}/\text{m}^3$ ，下降 77.3%。上海 2019 年与 2024 年 SO₂ 均为 7 $\mu\text{g}/\text{m}^3$ ，表明最后一个阶段按公布精度计算的净变化为 0。

矩阵比较揭示了城市、污染物和时间阶段三个层面的差异，也说明累计降幅相近并不意味着变化过程相同。阶段端点可能掩盖期间的年际反复，需与图 1 及原始浓度表结合分析。由于监测覆盖、来源切换和公布精度可能影响比较，本节结果仅作描述性分解，不直接归因于特定政策。

3.13 省级采暖期背景与六污染物差异

图11 采暖期与非采暖期的六污染物对比

黑龙江省13城市·2024年 | 非采暖期浓度=100

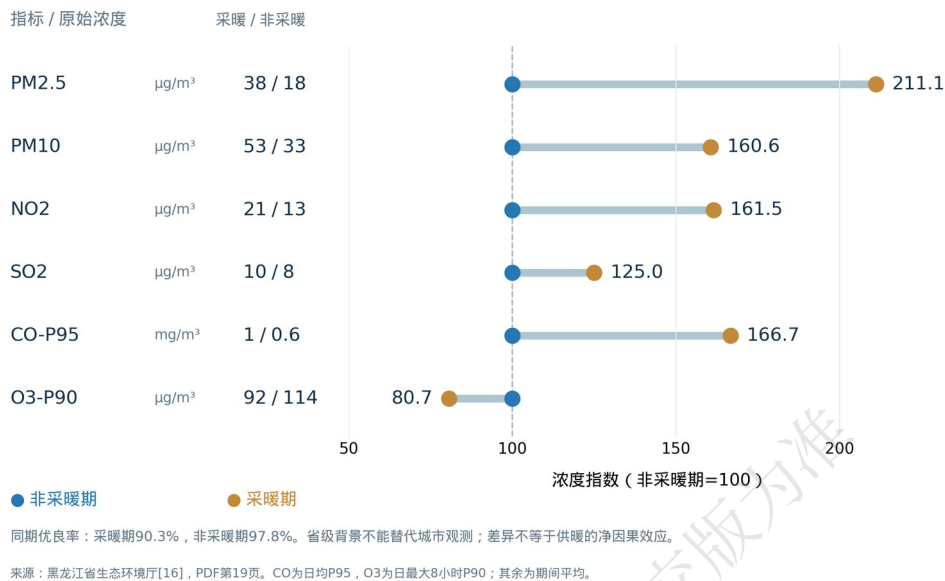


图 11 比较黑龙江省 13 城市 2024 年采暖期与非采暖期的六项污染物浓度[16]，用于描述哈尔滨所在省份的季节背景。以非采暖期为 100 计算采暖期相对指数，并同时列示原始浓度；CO 使用 mg/m^3 ，其余指标使用 $\mu\text{g}/\text{m}^3$ 。该统计范围为省内 13 城市，与哈尔滨市年度数据有所区别。

采暖期 PM2.5 为 $38 \mu\text{g}/\text{m}^3$ ，非采暖期为 $18 \mu\text{g}/\text{m}^3$ ，前者约为后者的 2.11 倍；PM10、NO2、SO2 和 CO 在采暖期也较高。O3 则由非采暖期的 114 降至采暖期的 $92 \mu\text{g}/\text{m}^3$ ，相对指数为 80.7。六项污染物的季节变化方向并不一致，其中臭氧与其余五项指标呈现不同特征。

采暖期与非采暖期的优良率分别为 90.3%和 97.8%，相差 7.5 个百分点。上述差异反映季节背景下的统计特征，但本研究未控制温度、日照、扩散条件及排放活动，尚不能分离供暖本身的影响。省级结果仅用于背景讨论，不用于替代哈尔滨缺失的城市观测。

3.14 气态污染物与空气质量等级专题

表 3 汇总了近期可获得的 CO 和 O3 记录。CO 采用 24 小时平均第 95 百分位浓度，O3 采用日最大 8 小时平均第 90 百分位浓度，两者均区别于年平均浓度。专题比较按相同指标和单位分别进行。

城市	年份	CO (mg/m ³)	O3 (μg/m ³)	来源
北京	2024	0.9	171	[28]
上海	2020	1.1	152	[6]
上海	2021	0.9	145	[7]
上海	2022	0.9	164	[8]
上海	2024	1.0	144	[10]
哈尔滨	2024	—	—	未取得可核验值

北京 2024 年 O3 为 171 μg/m³，上海为 144；CO 分别为 0.9 与 1.0 mg/m³[10][28]。两项指标的城市高低顺序相反。上海 2022 至 2024 年 O3 下降约 12.2%，CO 上升约 11.1%，进一步说明颗粒物改善并不能替代六项污染物的逐项检查。

表 4 上海空气质量等级天数（天）

年份	优	良	轻度	中度	重度	严重
2022	129	189	47	0	0	0
2024	132	191	38	3	1	0

2022 年与 2024 年上海空气质量等级均合计 365 天，优良天数分别为 318 天和 323 天 [8][10]。2024 年优良天数增加，但仍出现 3 天中度污染和 1 天重度污染，说明年度总体状况与较重污染过程需要分别评价。2024 年等级合计较日历天数少 1 天，原件未说明原因，本报告保留公布记录。

扩展指标：酸雨、扬尘、噪声与全国背景

表 5 上海酸雨与道路扬尘（仅列取得原件的年度）

年份	降水 pH	酸雨频率 (%)	道路扬尘 ($\mu\text{g}/\text{m}^3$)
2020	5.38	40.2	109
2021	5.56	26.4	81
2022	5.42	27.2	82
2024	5.46	26.9	88

来源[6][7][8][10]。2020 至 2024 年酸雨频率下降 13.3 个百分点，道路扬尘降低 21 $\mu\text{g}/\text{m}^3$ ；但 2022 至 2024 年扬尘由 82 升至 88。pH 是对数指标，不按普通浓度计算百分比变化；2023 年未核录，不插值补齐。

表 6 哈尔滨昼间平均等效声级（dB(A)）

年份	区域声环境	道路交通声环境	来源
2022	52.5	67.5	[13]
2023	53.0	67.1	[14]
2024	53.5	67.6	[15]

哈尔滨区域昼间声级从 52.5 升至 53.5 dB(A)；道路交通声级先降后升。分贝不按普通浓度相加或计算降幅。省报告另列哈尔滨 2024 年重度及以上污染 10 天[16]，是城市记录，区别于图 11 的省级均值。

全国背景[29]：2024 年地级及以上城市 PM_{2.5} 平均浓度 29.3 $\mu\text{g}/\text{m}^3$ 、优良天数比例 87.2%、重污染天数比例 0.9%，空气质量达标城市 222 个。全国汇总只作背景，不当作第四座城市、不与三市等权平均，也不用于推断全国由三市代表。

第四章 制作经验总结

4.1 主要难点与处理方法

本次制作最容易出错的是比例的分母与原文定位。上海 2024 年优良率应保留公报公布的 88.5%，不能把按 366 天计算得到的 88.3%当成原文取值，也不能无依据解释为沙尘扣除差异。处理方法是同时保留等级天数、比例和来源页码，逐项核对后再计算变化。

另一个问题是跨期可比性。新增十年资料中，早期浓度单位为 mg/m^3 ，后期为 $\mu\text{g}/\text{m}^3$ ，需先乘 1000 统一单位；2013 年前后评价指标发生变化，旧口径达标率不能直接与新口径优良率比较。北京 2023/2024 年公布浓度与官方同比的差异也保留说明，不倒推改数。

第三个问题是图表与结论的匹配。雷达图比较相对轮廓，云朵词云呈现选定资料词频，矩阵瀑布分解城市、污染物和阶段的交叉差异，哑铃图呈现省级季节背景。图形复杂度服务于信息层次，不使用三维透视夸大差异，也不把省级数值当作哈尔滨观测。

向前增加十年后，程序从 15 条年度记录扩展到 45 条，同时严格标记 9 条早期 PM2.5 缺失。图 1、图 5 采用 12 个有效年值，图 7 补充历史 PM10，图 3 和图 4 保留 2020—2024 年的专题窗口。历史达标天数保留原值，不为延长相关分析而强行换算比例。

4.2 收获与不足

本作品完成了从真实公开数据到可复现图表的基本流程。数据分析不仅是调用绘图函数，更包括判断数据能否放在一起比较、检查分母和单位、解释异常变化，以及对结论保持适当边界。把来源字段与每条记录绑定，也使后续修改和复核更容易。

不足之处是城市数量少、主要使用年度统计；新增省级采暖期资料只能提供季节背景，不能替代城市逐日观测。监测网络、评价标准、公布精度和来源可能变化。哈尔滨 2024 年 CO 与 O3 未获得可核验数值，保留缺失；未控制气象和排放活动，因此不提供治理因果效应。

后续可扩展到更多城市和逐日数据，并补充风速、降水、温度等因素。在参加数学建模或程序设计实践时，应先确定问题与数据粒度，再选择统计方法；保留原始数据、处理日志和完整代码，使图表和结论能够被他人复现。

4.3 结果要点

2013—2024 年，北京、上海、哈尔滨 PM2.5 公布值分别下降 65.7%、54.8%、50.6%；2020—2024 年对应降幅为 19.7%、12.5%、14.9%。2010—2012 年以 PM10 等记录补充，旧评价比例不与 AQI 优良率直接相比。长序列揭示总体改善，近五年专题分析揭示波动与污染物差异。

第五章 作品源代码与 AI 提示词截图

5.1 文件组织与运行说明

报告列出原始数据表、六个 Python 程序完整源码及 8 条 AI 辅助复现提示词。项目包含 29 项来源索引、12 张图（11 个图号、10 种类型）、13 份语料节选，以及完整词频、文档频次、固定词典和停用词表。图 8—11 同时提供 PNG 与 SVG。

文件或目录	作用
00_download.py	下载与记录摘要
01_prepare.py	原始表与扩展表的清洗、核验和统计
02_visualize.py	七张常规图表
03_corpus.py	13 份资料的提取、分词和计数
04_advanced.py	雷达、云朵词云、矩阵瀑布
05_seasonal.py	省级季节哑铃图
data/annual.csv、historical.csv	45 条年度记录、30 条重叠历史记录
data/pollutants.csv、primary_2022.csv	四项浓度端点、污染日构成
data/gas_extension.csv	6 条 CO 与 O3 记录
data/environment_extension.csv	23 条环境与背景统计
data/seasonal.csv、aqi_grades.csv	14 条季节记录、12 条等级天数
data/sources.json、data/corpus/	29 项来源、13 份节选与语料清单
raw/、results/、figures/	官方原件、核验结果、12 张图
requirements.txt、README.md	依赖与复现说明

运行步骤与数据核验补充

建议运行环境为 Python 3.10 及以上，依赖为 NumPy 1.24 及以上、Pandas 2.0 及以上、Matplotlib 3.9 及以上。安装命令：python -m pip install -r requirements.txt。

可选联网执行 00_download.py（需 curl）；离线依次执行 01_prepare.py、02_visualize.py、03_corpus.py、04_advanced.py、05_seasonal.py。先安装 requirements.txt 依赖并配置中文字体。下载失败保留日志；核录数据与包内官方快照支持离线复现。

源代码不修改原始 CSV，派生结果写入 results。2013 年 PM2.5 同比为空，因为前一年该字段缺失；pct_change 明确关闭填充。相关系数仅使用 2020—2024 年数据。其他未声明缺失或重复记录会使核验停止，以免带着错误绘图。

扩展数据质量检查

检查项	结果
气态污染物表	6 条；哈尔滨 2024 年 CO、O3 共 2 个空值
环境与季节表	23+14 条；省市范围与单位分开保留
空气质量等级表	12 条；2022 与 2024 各合计 365 天
矩阵瀑布	36 段；各面板基期加增减等于端点
语料计数	13 份；3843 词次；1338 词项
已声明缺失	不补零、不插值；不跨口径推算

补充原始表

城市	年份	首要污染物	天数	来源
上海	2022	O3	41	S22
上海	2022	PM2.5	6	S22

文件路径使用相对项目根目录，不写入个人电脑用户名。源码中的 source_id 通过 sources.json 映射到第六部分参考文献；更换数据时应同步更新来源信息。

程序1 官方资料下载 1/2

00_download.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
"""下载官方网页及其PDF附件；失败只记日志，不绕过访问限制。"""
from pathlib import Path
import subprocess
from urllib.parse import urljoin
from concurrent.futures import ThreadPoolExecutor
import hashlib
import json
import re
from datetime import datetime, timezone

ROOT = Path(__file__).resolve().parent
OUT = ROOT / 'raw'
OUT.mkdir(exist_ok=True)
sources = json.loads((ROOT / 'data/sources.json').read_text('utf-8'))

def fetch(item):
    key, source = item
    result = {'source_id': key, 'url': source['url'], 'files': []}
    result['retrieved_utc'] = datetime.now(timezone.utc).isoformat()
    try:
        body = subprocess.check_output(['curl', '-fsSL', '--max-time',
                                         '40', source['url']])
        is_pdf = body.startswith(b'%PDF')
        is_jpg = body.startswith(b'\xff\xd8')
        name = key + ('.pdf' if is_pdf else '.jpg' if is_jpg else '.html')
        (OUT / name).write_bytes(body)
        result['files'].append({'name': name, 'bytes': len(body),
                                'sha256': hashlib.sha256(body).hexdigest()})
    if not is_pdf and not is_jpg:
        html = body.decode('utf-8', errors='replace')
```

程序1 官方资料下载 2/2

00_download.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
links = re.findall(r'\"(?:(?:href|src)=\"([^\"]+\\.pdf)\"|\"')',
                    html, flags=re.I)
for i, link in enumerate(dict.fromkeys(links), start=1):
    url = urljoin(source['url'], link)
    # 对非 ASCII 网址编码, 保留已有转义和 URL 结构。
    from urllib.parse import quote
    url = quote(url, safe=':/%?=&')
    pdf = subprocess.check_output(['curl', '-fsSL',
                                    '--max-time', '40', url])
    if not pdf.startswith(b'%PDF'):
        raise ValueError('附件未返回 PDF 内容')
    name = f'{key}_attachment_{i}.pdf'
    (OUT / name).write_bytes(pdf)
    result['files'].append({'name': name, 'url': url,
                            'bytes': len(pdf),
                            'sha256': hashlib.sha256(pdf).hexdigest()})
    result['status'] = 'downloaded'
except Exception as error:
    result['status'] = 'failed_or_partial'
    result['error'] = str(error)
return result

if __name__ == '__main__':
    with ThreadPoolExecutor(max_workers=3) as pool:
        logs = list(pool.map(fetch, sources.items()))
    (OUT / 'download_log.json').write_text(
        json.dumps(logs, ensure_ascii=False, indent=2), encoding='utf-8')
    for log in logs:
        print(log['source_id'], log['status'], len(log['files']))
    print('下载文件仅作参考；数值需按报告定位核录和复核。')
```

程序2 数据与扩展核验 1/3

01_prepare.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
"""核验 2010–2024 年数据；显式保留早期 PM2.5 缺失与评价口径。"""
from pathlib import Path
import json
import numpy as np
import pandas as pd

ROOT = Path(__file__).resolve().parent
DATA, OUT = ROOT / 'data', ROOT / 'results'
OUT.mkdir(exist_ok=True)
CITIES = ['北京', '上海', '哈尔滨']
sources = json.loads((DATA / 'sources.json').read_text('utf-8'))

def read(name, keys, numeric, optional=()):
    df = pd.read_csv(DATA / name, encoding='utf-8-sig')
    for c in df.select_dtypes(include='object'):
        df[c] = df[c].str.strip()
    for c in numeric:
        df[c] = pd.to_numeric(df[c], errors='raise')
        v = df[c].dropna()
        assert np.isfinite(v).all() and (v >= 0).all(), (name, c)
    assert not df.drop(columns=list(optional)).isna().any().any(), name
    assert not df.duplicated(keys).any(), name
    assert set(df.source_id) <= set(sources), name
    return df.sort_values(keys).reset_index(drop=True)

annual = read('annual.csv', ['city', 'year'],
              ['year', 'pm25', 'good_rate'], ['pm25', 'good_rate'])
hist = read('historical.csv', ['city', 'year'],
            ['year', 'pm25', 'PM10', 'SO2', 'NO2', 'good_days',
             'CO_P95_mg_m3', 'O3_P90_ug_m3', 'old_good_rate'],
            ['pm25', 'CO_P95_mg_m3', 'O3_P90_ug_m3', 'old_good_rate'])
pol = read('pollutants.csv', ['city', 'year'],
           ['year', 'PM2.5', 'PM10', 'NO2', 'SO2'], ['PM10'])
primary = read('primary_2022.csv', ['city', 'year', 'pollutant'],
               ['year', 'days'])
assert set(zip(annual.city, annual.year)) == {
    (c, y) for c in CITIES for y in range(2010, 2025)}
assert set(zip(hist.city, hist.year)) == {
    (c, y) for c in CITIES for y in range(2010, 2020)}
assert annual.pm25.isna().equals(annual.year.lt(2013))
assert annual.good_rate.isna().equals(annual.year.lt(2020))
```

程序2 数据与扩展核验 2/3

01_prepare.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
assert annual.good_rate.dropna().between(0, 100).all()
for c in ['pm25', 'CO_P95_mg_m3', 'O3_P90_ug_m3']:
    assert hist[c].isna().equals(hist.year.lt(2013))
assert hist.old_good_rate.isna().equals(hist.year.ge(2013))
assert hist.old_good_rate.dropna().between(0, 100).all()
assert hist.good_days.between(0, 366).all()
assert (hist.dropna(subset=['pm25']).pm25 <=
        hist.dropna(subset=['pm25']).PM10).all()
assert (pol.dropna(subset=['PM10'])['PM2.5'] <=
        pol.dropna(subset=['PM10']).PM10).all()
assert set(zip(pol.city, pol.year)) == {
    (c, y) for c in CITIES for y in [2020, 2024]}
assert set(map(tuple, pol.loc[pol.PM10.isna(), ['city', 'year']].values)) == {
    ('哈尔滨', 2020)}
assert set(pol.pm10_source_id) <= set(sources)
assert primary.days.sum() == 47
for other, col in [(pol, 'PM2.5'), (hist, 'pm25')]:
    check = other.merge(annual, on=['city', 'year'], suffixes=('_x', '_y'))
    left = check[col] if col == 'PM2.5' else check.pm25_x
    right = check.pm25 if col == 'PM2.5' else check.pm25_y
    assert np.allclose(left, right, equal_nan=True)
# 不补齐PM2.5, 不把旧口径达标率并入AQI优良率。
annual['pm25_yoy_pct'] = annual.groupby('city').pm25.transform(
    lambda x: x.pct_change(fill_method=None) * 100)
annual['good_change_pp'] = annual.groupby('city').good_rate.diff()
summary = []
for city, group in annual.groupby('city'):
    g = group.dropna(subset=['pm25'])
    recent = group[group.year >= 2020]
    first, last = g.iloc[0], g.iloc[-1]
    summary.append(dict(city=city, n=len(g), pm25_mean=g.pm25.mean(),
        pm25_median=g.pm25.median(), pm25_min=g.pm25.min(),
        pm25_max=g.pm25.max(), pm25_reduction_pct=
        (first.pm25-last.pm25)/first.pm25*100,
        recent_reduction_pct=(recent.iloc[0].pm25-last.pm25)
        / recent.iloc[0].pm25*100,
        recent_good_gain_pp=last.good_rate-recent.iloc[0].good_rate,
        recent_pearson_r=recent.pm25.corr(recent.good_rate)))
summary = pd.DataFrame(summary)
for name, table in [('annual_clean', annual), ('historical_clean', hist),
    ('pollutants_clean', pol), ('primary_clean', primary),
```

程序2 数据与扩展核验 3/3

01_prepare.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
( 'summary', summary)]:
    table.to_csv(OUT / f'{name}.csv', index=False, encoding='utf-8-sig',
                 float_format='%.6f')
audit = dict(annual_rows=len(annual), historical_rows=len(hist),
             endpoint_rows=len(pol), primary_rows=len(primary),
             early_pm25_missing=int(annual.pm25.isna().sum()),
             historical_rate_not_joined=int(annual.good_rate.isna().sum()),
             endpoint_pm10_missing=int(pol.PM10.isna().sum()),
             duplicates=0, imputed=0, pm25_period='2013-2024',
             rate_comparison_period='2020-2024',
             note='上海 2024 用公报 88.5%；哈尔滨优良率口径见 README。')
(OUT / 'quality_log.json').write_text(
    json.dumps(audit, ensure_ascii=False, indent=2), encoding='utf-8')
print(summary.round(4).to_string(index=False))
print('核验通过：45 条年度记录，PM2.5 有效 36 条；未插补。')

# 扩展专题保留地域范围，不能把省级数据替换成城市数据。
gas = read('gas_extension.csv', ['city', 'year'],
          ['year', 'CO_P95_mg_m3', '03_P90_ug_m3'],
          ['CO_P95_mg_m3', '03_P90_ug_m3'])
env = read('environment_extension.csv', ['scope', 'year', 'metric'],
          ['year', 'value'])
season = read('seasonal.csv', ['scope', 'year', 'period', 'metric'],
             ['year', 'value'])
grades = read('aqi_grades.csv', ['city', 'year', 'grade'],
             ['year', 'days'])
assert gas.CO_P95_mg_m3.isna().equals(gas.city.eq('哈尔滨'))
assert gas.03_P90_ug_m3.isna().equals(gas.city.eq('哈尔滨'))
assert grades.groupby('year').days.sum().eq(365).all()
assert env.loc[env.unit.eq('%'), 'value'].between(0, 100).all()
assert set(season.scope) == {'黑龙江省 13 城市'}
for name, table in [('gas_extension', gas), ('environment_extension', env),
                   ('seasonal', season), ('aqi_grades', grades)]:
    table.to_csv(OUT / (name + '_clean.csv'), index=False)
audit['extension_rows'] = dict(gas=len(gas), environment=len(env),
                              seasonal=len(season), grades=len(grades))
audit['gas_missing_cells'] = int(gas[
    ['CO_P95_mg_m3', '03_P90_ug_m3']].isna().sum().sum())
(OUT / 'quality_log.json').write_text(
    json.dumps(audit, ensure_ascii=False, indent=2), encoding='utf-8')
```

程序3 七张常规图表 1/6

02_visualize.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
"""读取核验结果，用Matplotlib生成七张、六种图表。"""
from pathlib import Path
from io import BytesIO
import os
import numpy as np
import pandas as pd
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
from matplotlib import font_manager

ROOT = Path(__file__).resolve().parent
OUT = ROOT / 'figures'
OUT.mkdir(exist_ok=True)
# 可通过环境变量指定字体；否则寻找本机常用中文字体。
font_path = os.getenv('CHINESE_FONT')
if font_path:
    font_manager.fontManager.addfont(font_path)
    font_name = font_manager.FontProperties(fname=font_path).get_name()
else:
    installed = {f.name for f in font_manager.fontManager.ttflist}
    choices = ['Microsoft YaHei', 'SimHei', 'PingFang SC',
               'Noto Sans CJK SC', 'Droid Sans Fallback']
    font_name = next((x for x in choices if x in installed), None)
    if font_name is None:
        raise RuntimeError('请安装中文字体，或设置 CHINESE_FONT 为字体文件路径')
plt.rcParams.update({'font.family': [font_name, 'DejaVu Sans'],
                    'axes.unicode_minus': False, 'font.size': 11,
                    'axes.spines.top': False, 'axes.spines.right': False,
                    'figure.facecolor': 'white', 'savefig.facecolor': 'white'})
a = pd.read_csv(ROOT / 'results/annual_clean.csv')
p = pd.read_csv(ROOT / 'results/pollutants_clean.csv')
z = pd.read_csv(ROOT / 'results/primary_clean.csv')
h = pd.read_csv(ROOT / 'results/historical_clean.csv')
colors = {'北京': '#256B9A', '上海': '#D27B35', '哈尔滨': '#34836D'}
cities = ['北京', '上海', '哈尔滨']
source = '来源：京沪生态环境局、哈尔滨市统计局及黑龙江省生态环境厅[1]–[16]。'

def save_png(fig, path, dpi=300):
    buffer = BytesIO()
```

程序3 七张常规图表 2/6

02_visualize.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
fig.savefig(buffer, format='png', dpi=dpi)
path.write_bytes(buffer.getvalue())

def save(fig, name, note=source):
    fig.text(.07, .025, note, fontsize=8, color='#444444')
    fig.tight_layout(rect=(0, .07, 1, 1))
    save_png(fig, OUT / name, dpi=220)
    plt.close(fig)

# 1 长期 PM2.5: 早期没有该指标, 不插补。
fig, ax = plt.subplots(figsize=(8.4, 4.5))
for city in cities:
    g = a[(a.city == city) & (a.year >= 2013)]
    ax.plot(g.year, g.pm25, '-o', color=colors[city], label=city, lw=2)
    for row in g[g.year.isin([2013, 2024])].itertuples():
        dy = -16 if city == '上海' else 8
        ax.annotate(f'{row.pm25:g}', (row.year, row.pm25),
                    xytext=(0, dy), textcoords='offset points',
                    ha='center', fontsize=10)
ax.axvspan(2010, 2012.6, color='#eeeeee', alpha=.8)
ax.text(2011.25, 58, '原表未列 PM2.5\n不插补', ha='center', fontsize=10)
ax.set(xticks=range(2010, 2025, 2), xlim=(2009.8, 2024.7),
       ylim=(18, 100), xlabel='年份', ylabel='PM2.5 年均浓度 (μg/m³)',
       title='图1 2013–2024 年三市 PM2.5 变化')
ax.grid(alpha=.2)
ax.legend(frameon=False)
save(fig, '01_line.png',
      '来源: 国家统计局[20]–[26]、三市官方公报[1]–[15]; 2010–2012 年原表未列 PM2.5。')

# 2 六面板柱状图: 每项独立刻度, 保留百分位与单位差别。
gas = pd.read_csv(ROOT / 'data/gas_extension.csv')
fig, axes = plt.subplots(2, 3, figsize=(10, 6.4))
fig.subplots_adjust(left=.07, right=.96, top=.80, bottom=.18,
                    wspace=.32, hspace=.72)
fig.text(.055, .95, '图2 2024 年六项污染物比较',
        fontsize=18, weight='bold', color='#16324B')
fig.text(.055, .89, '同项比较城市; 各面板独立刻度, 不能跨面板比较柱高。',
        fontsize=10, color='#637788')
cols = ['PM2.5', 'PM10', 'NO2', 'SO2']
for j, (metric, ax) in enumerate(zip(cols+['CO', 'O3'], axes.flat)):
```

程序3 七张常规图表 3/6

02_visualize.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
values = []
for city in cities:
    if j < 4:
        v = p.loc[(p.city==city)&(p.year==2024), metric].iloc[0]
    else:
        field = 'CO_P95_mg_m3' if j==4 else 'O3_P90_ug_m3'
        v = gas.loc[(gas.city==city)&(gas.year==2024), field].iloc[0]
    values.append(v)
upper = max(v for v in values if pd.notna(v))*1.32
bars = ax.bar(range(3), values, .48,
              color=[colors[c] for c in cities], zorder=3)
for k, v in enumerate(values):
    if pd.notna(v):
        ax.text(k, v+upper*.035, f'{v:g}', ha='center', fontsize=11)
    else:
        ax.text(k, upper*.08, '未取得', ha='center',
                fontsize=9, color='#7A8995')
unit = 'mg/m³' if j==4 else 'µg/m³'
basis = '年均' if j<4 else ('24h·P95' if j==4 else '8h最大·P90')
ax.set_title(f'{metric} | {basis} ({unit}) ',
             loc='left', fontsize=11, pad=12, color='#16324B')
ax.set(xticks=range(3), xticklabels=cities, ylim=(0, upper))
ax.grid(axis='y', color='#E3EAF0', linewidth=.7)
ax.set_axisbelow(True)
ax.spines[['left', 'bottom']].set_color('#D4DEE5')
ax.tick_params(length=0, labelsize=9)
fig.text(.055, .08, 'CO与O3为百分位浓度；哈尔滨两项缺值不补零。',
        fontsize=10, color='#637788')
fig.text(.055, .035, '来源：2024年官方公报[5][10][15][16][28]。',
        fontsize=8, color='#637788')
save_png(fig, OUT/'02_bar.png')
plt.close(fig)

# 3 热力图：统一基期的变化率；负值表示浓度下降。
changes = []
for city in cities:
    g = p[p.city == city].set_index('year')[cols]
    changes.append((g.loc[2024] / g.loc[2020] - 1) * 100).values)
m = np.array(changes)
fig, ax = plt.subplots(figsize=(8.4, 4.0))
```

程序3 七张常规图表 4/6

02_visualize.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
cmap = plt.get_cmap('RdBu_r').copy()
cmap.set_bad('#E5E7EB')
im = ax.imshow(m, cmap=cmap, vmin=-50, vmax=50, aspect='auto')
for i in range(len(cities)):
    for j in range(4):
        label = '未列示' if np.isnan(m[i, j]) else f'{m[i,j]:+.1f}%'
        ax.text(j, i, label, ha='center', va='center', fontsize=13)
ax.set(xticks=range(4), xticklabels=cols, yticks=range(len(cities)),
       yticklabels=cities, title='图3 2024年相对2020年的浓度变化率')
fig.colorbar(im, ax=ax, label='变化率 (%) , 负值表示改善')
save(fig, '03_heatmap.png',
     source + '\n'
     '起止年变化率由程序计算；灰格表示缺少可核验PM10年值。')

# 4 散点图：每点为一个城市年度，不拟合因果模型。
fig, ax = plt.subplots(figsize=(8.4, 4.5))
for city in cities:
    g = a[(a.city == city) & (a.year >= 2020)]
    ax.scatter(g.pm25, g.good_rate, s=65, color=colors[city], label=city)
    for row in g.itertuples():
        offset = (-25, -13) if row.year == 2023 else (5, 6)
        if city == '北京' and row.year == 2022:
            offset = (-28, -12)
        if city == '哈尔滨' and row.year == 2021:
            offset = (-35, 8)
        if city == '哈尔滨' and row.year == 2022:
            offset = (6, -13)
        ax.annotate(str(row.year), (row.pm25, row.good_rate),
                   xytext=offset, textcoords='offset points', fontsize=9)
ax.set(xlim=(23, 51), ylim=(71, 95), xlabel='PM2.5 年均浓度 (μg/m³)',
       ylabel='优良天数占比 (%)', title='图4 2020-2024年PM2.5与优良天数占比')
ax.grid(alpha=.2)
ax.legend(frameon=False)
save(fig, '04_scatter.png',
     source + '\n'
     '哈尔滨2020为303/366折算；2024公报分母存在不一致，详见正文。')

# 5 箱线图叠加原始点：仅描述12个年度值的离散程度。
fig, ax = plt.subplots(figsize=(8.4, 4.5))
series = [a[a.city == c].pm25.dropna().to_numpy() for c in cities]
```

程序3 七张常规图表 5/6

02_visualize.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
bp = ax.boxplot(series, tick_labels=cities, widths=.45,
                patch_artist=True, showfliers=False)
for box, city in zip(bp['boxes'], cities):
    box.set(facecolor=colors[city], alpha=.25)
for i, (city, values) in enumerate(zip(cities, series), start=1):
    ax.scatter(i + np.linspace(-.12, .12, len(values)), values,
               color=colors[city], s=45, zorder=3)
ax.set(ylim=(20, 98), ylabel='PM2.5 年均浓度 (μg/m³)',
       title='图5 2013–2024 年 PM2.5 年度值分布')
ax.grid(axis='y', alpha=.2)
save(fig, '05_box.png',
     '来源: 国家统计局[20]–[26]及三市公报[1]–[15]。\\n'
     '每市 n=12; 散点保留全部年值, 不能解释为日浓度分布。')

# 6 饼图: 分母是 47 个污染日, 不是全年 365 天。
fig = plt.figure(figsize=(10, 6.3))
ax = fig.add_axes([.04, .30, .46, .48])
z = z.set_index('pollutant').loc[['03', 'PM2.5']]
ax.pie(z.days, labels=['03\\n41 天', 'PM2.5\\n6 天'],
      autopct='%.1f%', startangle=90, counterclock=False,
      colors=['#D27B35', '#256B9A'], pctdistance=.7,
      wedgeprops={'edgecolor': 'white', 'linewidth': 2, 'width': .40})
ax.text(0, 0, '47 天\\n污染日', ha='center', va='center', fontsize=13)
ax.set_title('2022 年首要污染物', fontsize=12, pad=12)
fig.text(.055, .95, '图6 上海污染日构成与空气质量等级',
        fontsize=18, weight='bold', color='#16324B')
fig.text(.055, .89, '左图分母为污染日; 右图分母为公报列示的全年等级合计天数。',
        fontsize=10, color='#637788')
grades = pd.read_csv(ROOT/'data/aqi_grades.csv')
names = ['优', '良', '轻度污染', '中度污染', '重度污染', '严重污染']
palette = ['#368C83', '#9BC8AB', '#E9BD68', '#DE8E58', '#B75F69', '#7D5A82']
ax = fig.add_axes([.58, .39, .35, .32])
for i, year in enumerate([2022, 2024]):
    counts = grades[grades.year==year].set_index('grade').loc[names, 'days']
    left = 0
    for n, color in zip(names, palette):
        value = counts[n]/counts.sum()*100
        ax.barh(i, value, left=left, height=.36, color=color,
                label=n if i==0 else None)
        if value>8:
```

程序3 七张常规图表 6/6

02_visualize.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
ax.text(left+value/2, i, str(counts[n]),
        ha='center', va='center', fontsize=10)
left += value
ax.set(xlim=(0, 100), ylim=(1.65, -.65), yticks=[0,1],
       yticklabels=['2022', '2024'], xticks=[0,50,100],
       xticklabels=['0%', '50%', '100%'])
ax.set_title('六类等级天数构成', loc='left', fontsize=12, pad=15)
ax.spines[['top','bottom','left','right']].set_visible(False)
ax.tick_params(length=0)
ax.legend(ncol=3, loc='upper center', bbox_to_anchor=(.5,-.25),
         fontsize=8, frameon=False, columnspacing=.9, handlelength=1)
fig.text(.055, .17, '2024年: 优132、良191、轻度38、中度3、重度1、严重0天。',
        fontsize=10, color='#637788')
fig.text(.055, .11, '两年等级均合计365天; 不补入2024年与日历天数相差的1天。',
        fontsize=10, color='#637788')
fig.text(.055, .035, '来源: 上海2022、2024年生态环境状况公报[8][10]。',
        fontsize=8, color='#637788')
save_png(fig, OUT/'06_pie.png')
plt.close(fig)
# 7 历史PM10补充: 2012/2013处断开, 提示监测评价体系变更。
fig, ax = plt.subplots(figsize=(8.4, 4.5))
for city in cities:
    for start, end in [(2010, 2012), (2013, 2019)]:
        g = h[(h.city == city) & h.year.between(start, end)]
        ax.plot(g.year, g.PM10, '-o', lw=2, color=colors[city],
               label=city if start == 2010 else None)
ax.axvline(2012.5, ls='--', lw=1, color='#888888')
ax.text(2012.6, 123, '监测评价体系变更', fontsize=9)
ax.set(xticks=range(2010, 2020), ylim=(35, 130),
       xlabel='年份', ylabel='PM10 年均浓度 (μg/m³)',
       title='图7 2010–2019年PM10历史记录')
ax.grid(alpha=.2)
ax.legend(frameon=False, loc='lower left')
save(fig, '07_history.png',
     '来源: 国家统计局[17]–[26]; 早期mg/m³ 乘1000统一单位。\\n'
     '2012/2013处不连线; 浓度历史展示不等于新旧空气质量评价结果可直接相比。')
print('已生成7张、6种图表, 缺失值未补零。')
```

程序4 扩展语料与词频 1/3

03_corpus.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
"""从官方原件提取相关正文，保留语料、词频及统计口径。"""
from pathlib import Path
from collections import Counter
import hashlib
import json
import re
import fitz
import jieba
import pandas as pd
from lxml import html

ROOT = Path(__file__).resolve().parent
RAW = ROOT / 'raw'
OUT = ROOT / 'results'
CORPUS = ROOT / 'data/corpus'
CORPUS.mkdir(exist_ok=True)

def paragraphs(filename, xpath):
    tree = html.fromstring((RAW / filename).read_bytes())
    nodes = tree.xpath(xpath)
    assert nodes, filename
    return '\n'.join(''.join(n.itertext()).strip() for n in nodes)

texts = {}
texts['B21'] = paragraphs('B21.html', '//*[@id="h_dl_c"]//p')
texts['B24'] = paragraphs('B24.html', '//*[@id="h_dl_c"]//p')
texts['STD13'] = paragraphs('STD13.html',
    '//div[contains(@class,"neiright_JPZ_GK_CP")]//p')
for key in ['S20', 'S21']:
    t = paragraphs(key + '.html', '//*[@id="zoom"]//p')
    texts[key] = t.split('一、', 1)[1].split('二、', 1)[0]
t = paragraphs('S23.html', '//*[@id="zoom"]//p')
texts['S23'] = t.split('2023 年，上海市环境空气质量指数', 1)[1]
texts['S23'] = '2023 年，上海市环境空气质量指数' + texts['S23']
```

程序4 扩展语料与词频 2/3

03_corpus.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
for key, pages in [('S22', [4, 5, 6]), ('S24', [5, 6])]:
    passages = []
    with fitz.open(RAW / (key + '.pdf')) as doc:
        for page in pages:
            for block in doc[page].get_text('blocks'):
                t = re.sub(r'\s+', '', block[4])
                if '。' in t and len(re.findall(r'[\u4e00-\u9fff]', t)) > 25:
                    passages.append(t)
            texts[key] = '\n'.join(passages)
for key, page in [('H22', 22), ('H23', 18), ('H24', 18)]:
    with fitz.open(RAW / (key + '_attachment_1.pdf')) as doc:
        t = re.sub(r'\s+', '', doc[page].get_text())
        start = t.index('空气质量')
        texts[key] = t[start:].split('安全生产')[0]
texts['CH24_release'] = paragraphs('CH24_release.html',
    '//div[@class="TRS_Editor"]')
texts['P24'] = (CORPUS / 'P24_manual.txt').read_text()

# 固定词典将领域术语作为完整词项，不手工调整词频。
terms = ('空气质量 环境空气质量 细颗粒物 可吸入颗粒物 '
    '二氧化硫 二氧化氮 臭氧 一氧化碳 重污染天气 '
    '大气污染防治 年均浓度 优良天数 京津冀 长三角 '
    '生态环境 环境质量 酸雨频率 道路扬尘 采暖期 '
    '非采暖期 声环境 地表水 水质断面').split()
for term in terms:
    jieba.add_word(term, freq=100000)
stop = set(('的 了 和 与 及 或 是 为 在 对 将 从 中 其 以 由 '
    '等 把 按 也 都 更 最 已 各 可 要 有 个 上 下 全 市 我 们 '
    '记者 部分 其中 同时 通过 进行 进一步 以来 以及 目前 '
    '达到 表明 近年 年 年度 年份 月 日 年均值 微克 立方米 '
    '毫克 百分点 百分位数 第 最大 平均值 监测数据').split())
(CORPUS / 'stopwords.txt').write_text('\n'.join(sorted(stop)), encoding='utf-8')
(CORPUS / 'terms.txt').write_text('\n'.join(terms), encoding='utf-8')
counts = Counter()
```

程序4 扩展语料与词频 3/3

03_corpus.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
document_counts = Counter()
manifest = []
seen = set()
for key, text in texts.items():
    text = re.sub(r'[\t\r]+', ' ', text).strip()
    lines = [line.strip() for line in text.splitlines() if line.strip()]
    selected = []
    for line in lines:
        if line not in seen:
            seen.add(line)
            selected.append(line)
    text = '\n'.join(selected)
    (CORPUS / f'{key}.txt').write_text(text, encoding='utf-8')
    tokens = [w for w in jieba.lcut(text)
               if re.fullmatch(r'[\u4e00-\u9fff]{2,}', w) and w not in stop]
    counts.update(tokens)
    document_counts.update(set(tokens))
    manifest.append({'source_id': key, 'characters': len(text),
                    'retained_tokens': len(tokens),
                    'sha256': hashlib.sha256(text.encode()).hexdigest()})
total = sum(counts.values())
rows = sorted(counts.items(), key=lambda p: (-p[1], p[0]))
pd.DataFrame([(w, c, c / total * 100, document_counts[w]) for w, c in rows],
              columns=['word', 'count', 'frequency_pct', 'document_count']).to_csv(
    OUT / 'word_frequency.csv', index=False, encoding='utf-8-sig')
(CORPUS / 'manifest.json').write_text(json.dumps({
    'sources': manifest, 'total_tokens': total, 'unique_terms': len(counts),
    'scope': 'B21/B24 正文; S20/S21 质量章节; S23 质量概述; S22 PDF 第 5–7 页、S24 第 6–7 页完整叙述
块; H22 第 23 页、H23/H24 第 19 页空气至水质段; STD13 访谈; 全国 2024 发布稿正文; P24 第 19 页采暖期两段
人工转录。',
    'method': '跨资料精确段落去重; jieba 精确分词; 保留至少两字纯中文词; 固定词典和停用词; 不合并同义
词; 展示前 150 词; 文档频次另列。',
    'caution': '篇幅不均; 词频仅描述选定语料, 不能代表政策重要性或污染程度。'
}), ensure_ascii=False, indent=2), encoding='utf-8')
print('Corpus:', total, 'tokens;', len(counts), 'terms')
```

程序5 雷达、云朵词云与矩阵瀑布 1/6

04_advanced.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
"""生成雷达图、云朵词云与多污染物分阶段矩阵瀑布图。"""
from pathlib import Path
from io import BytesIO
import os
import numpy as np
import pandas as pd
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
from matplotlib import font_manager
from matplotlib.patches import FancyBboxPatch
from wordcloud import WordCloud

ROOT = Path(__file__).resolve().parent
OUT = ROOT / 'figures'
font_path = os.getenv('CHINESE_FONT')
if not font_path:
    for f in font_manager.fontManager.ttflist:
        if f.name in ['Microsoft YaHei', 'SimHei',
                      'Noto Sans CJK SC', 'Droid Sans Fallback']:
            font_path = f.fname
            break
if not font_path:
    raise RuntimeError('请设置 CHINESE_FONT 为中文字体文件路径')
font_manager.fontManager.addfont(font_path)
font_name = font_manager.FontProperties(fname=font_path).get_name()
INK, MUTED, GRID = '#16324B', '#637788', '#DBE3EA'
COLORS = ['#2878B5', '#E58B34', '#36936D']
plt.rcParams.update({'font.family': [font_name, 'DejaVu Sans'],
                    'axes.unicode_minus': False, 'font.size': 11,
                    'text.color': INK, 'axes.labelcolor': INK,
                    'xtick.color': MUTED, 'ytick.color': MUTED,
                    'figure.facecolor': 'white', 'savefig.facecolor': 'white'})
cities = ['北京', '上海', '哈尔滨']

def title(fig, number, main, sub):
    fig.text(.055, .955, f'图{number} {main}', fontsize=18,
            va='top', weight='bold')
    fig.text(.055, .897, sub, fontsize=10, color=MUTED, va='top')
```

程序5 雷达、云朵词云与矩阵瀑布 2/6

04_advanced.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
def save(fig, filename, source):
    fig.text(.055, .035, source, fontsize=8, color=MUTED)
    buffer = BytesIO()
    fig.savefig(buffer, format='png', dpi=300, facecolor='white')
    (OUT / filename).write_bytes(buffer.getvalue())
    fig.savefig(OUT / filename.replace('.png', '.svg'))
    plt.close(fig)

# 雷达：每一城市每一污染物均以本市 2013 年为 100。
h = pd.read_csv(ROOT / 'data/historical.csv')
p = pd.read_csv(ROOT / 'data/pollutants.csv')
gas = pd.read_csv(ROOT / 'data/gas_extension.csv')
cols = ['PM2.5', 'PM10', 'NO2', 'SO2', 'CO', 'O3']
angles = np.arange(6) * np.pi / 3
closed = np.r_[angles, angles[0]]
fig = plt.figure(figsize=(10, 6.1))
title(fig, 8, '六项污染物的长期变化轮廓',
      '2024 年相对本市 2013 年的浓度指数 | 基期=100; 数值越小, 累计降幅越大')
records = []
for i, (city, color) in enumerate(zip(cities, COLORS)):
    baseline = h[(h.city == city) & (h.year == 2013)].iloc[0]
    end = p[(p.city == city) & (p.year == 2024)].iloc[0]
    g = gas[(gas.city == city) & (gas.year == 2024)].iloc[0]
    base = np.array([baseline.pm25, baseline.PM10,
                    baseline.NO2, baseline.SO2, baseline.CO_P95_mg_m3,
                    baseline.O3_P90_ug_m3])
    values = np.r_[end[cols[:4]].to_numpy(dtype=float),
                  g.CO_P95_mg_m3, g.O3_P90_ug_m3]
    ratio = values / base * 100
    for j, name in enumerate(cols):
        records.append([city, name, base[j], values[j], ratio[j]])
    ax = fig.add_axes([.075 + i * .32, .30, .21, .41], polar=True)
    ax.set_theta_offset(np.pi / 2)
    ax.set_theta_direction(-1)
    ax.set_ylim(0, 105)
    ax.set_yticks([25, 50, 75, 100])
    ax.set_yticklabels([])
    ax.set_rlabel_position(32)
    ax.set_xticks(angles)
    ax.set_xticklabels(cols, fontsize=11)
```

程序5 雷达、云朵词云与矩阵瀑布 3/6

04_advanced.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
ax.tick_params(axis='x', pad=10)
ax.grid(color=GRID, linewidth=.7)
ax.spines['polar'].set_visible(False)
ax.plot(closed, np.ones(7) * 100, color='#A6B5C1',
        ls='--', lw=1)
ax.plot(closed, np.r_[ratio, ratio[0]], color=color, lw=2)
if np.isfinite(ratio).all():
    ax.fill(closed, np.r_[ratio, ratio[0]], color=color, alpha=.12)
ax.scatter(angles, ratio, color=color, s=28, zorder=3)
offsets = [(0,9), (8,4), (8,-5), (0,-14), (-8,-6), (7,-12)]
for j, (a, r) in enumerate(zip(angles, ratio)):
    if not np.isfinite(r):
        ax.text(a, 70, '未取得', ha='center', va='center',
                fontsize=8, color=MUTED)
        continue
    align = 'center' if j in [0,3] else ('right' if j==4 else 'left')
    ax.annotate(f'{r:.1f}', (a, r), xytext=offsets[j],
               textcoords='offset points', ha=align,
               fontsize=10, color=color)
fig.text(.18 + i * .32, .79, city, ha='center',
        fontsize=14, color=color, weight='bold')
fig.text(.055, .225, '同心圆由内向外为 25、50、75、100；虚线六边形为基期 100。',
        fontsize=10, color=MUTED)
fig.text(.055, .175, 'CO 采用 24 小时第 95 百分位；O3 采用日最大 8 小时第 90 百分位。',
        fontsize=10, color=MUTED)
fig.text(.055, .125, '哈尔滨两项缺值保留断线；不补零、不计算多边形综合评分。',
        fontsize=10, color=MUTED)
save(fig, '08_radar.png',
      '来源：国家统计局[20]及 2024 年官方公报[5][10][15][16][28]；各项独立以 2013=100。')
pd.DataFrame(records, columns=['city', 'pollutant', 'base_2013',
                              'value_2024', 'index_2024']).to_csv(
    ROOT / 'results/radar_values.csv', index=False)

# 词云：云朵掩膜只决定边界，不修改真实词频。
import json
manifest = json.loads((ROOT / 'data/corpus/manifest.json').read_text())
freq = pd.read_csv(ROOT / 'results/word_frequency.csv')
top = freq.head(150)
palette = ['#16324B', '#2878B5', '#237E81', '#36936D', '#B98435']
def word_color(word, **kwargs):
```

程序5 雷达、云朵词云与矩阵瀑布 4/6

04_advanced.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
    return palette[sum(map(ord, word)) % len(palette)]
yy, xx = np.mgrid[0:1100, 0:1900]
inside = np.zeros_like(xx, dtype=bool)
for cx, cy, rx, ry in [(350,710,270,240), (570,535,300,290),
                        (930,350,325,310), (1270,520,305,290),
                        (1540,700,280,250), (950,785,730,175)]:
    inside |= ((xx-cx)/rx)**2 + ((yy-cy)/ry)**2 <= 1
mask = np.where(inside, 0, 255).astype('uint8')
cloud = WordCloud(font_path=font_path, mask=mask,
                  background_color='white', max_words=150, prefer_horizontal=1,
                  relative_scaling=.45, random_state=28, margin=7,
                  min_font_size=13, max_font_size=125, color_func=word_color,
                  contour_width=1, contour_color='#CBDDE0'
                  ).generate_from_frequencies(dict(zip(top.word, top['count'])))
fig = plt.figure(figsize=(10, 8))
title(fig, 9, '生态环境资料中的关键词云',
      f'{len(manifest["sources"])}份官方资料节选 | '
      f'{manifest["total_tokens"]:,}个有效词次 | 真实词频前 150 项')
ax = fig.add_axes([.035, .26, .93, .57])
ax.imshow(cloud, interpolation='bilinear')
ax.axis('off')
fig.text(.065, .265, '高频词 TOP 8', fontsize=11, weight='bold')
for col in range(2):
    # 卡片与计数条采用圆角；条长仍严格按实际词频线性编码。
    left = .055 + col*.46
    card = FancyBboxPatch((left, .115), .43, .13,
                          boxstyle='round,pad=0.008,rounding_size=0.012',
                          transform=fig.transFigure, facecolor='#F1F6F7',
                          edgecolor='none', zorder=0)
    fig.add_artist(card)
    ax = fig.add_axes([left+.095, .125, .29, .11])
    r = freq.iloc[col*4:col*4+4]
    ax.set_facecolor('none')
    limit = freq['count'].max()*1.2
    for i, row in enumerate(r.itertuples()):
        # 数据坐标的纵横比例换算，保证两端在显示空间中圆。
        aspect = (limit/2.9)/(4/.88)
        bar = FancyBboxPatch((0, i-.25), row.count, .5,
                              boxstyle=f'round,pad=0,rounding_size={.25*aspect}',
                              mutation_aspect=1/aspect, facecolor='#237E81',
```

程序5 雷达、云朵词云与矩阵瀑布 5/6

04_advanced.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
        edgecolor='none')
    ax.add_patch(bar)
    ax.set_yticks(range(4), r.word, fontsize=11)
    ax.set_ylim(3.5, -.5)
    ax.set_xlim(0, limit)
    for i, row in enumerate(r.itertuples()):
        ax.text(row.count+1, i, f'{row.count}', va='center', fontsize=11)
    ax.set_xticks([])
    for spine in ax.spines.values():
        spine.set_visible(False)
    ax.tick_params(length=0)
fig.text(.055, .075,
        '字体大小编码词频；颜色、位置及云朵轮廓仅作排版。下方为前8词的实际计数。',
        fontsize=9, color=MUTED)
save(fig, '09_wordcloud.png',
        '语料覆盖空气、酸雨、扬尘及其他生态环境统计；来源明细及文档频次随项目提供。')

# 矩阵式瀑布：3城市×4指标×3阶段，全部以本市本项2013=100。
stages = [2013, 2016, 2019, 2024]
records = []
for part, metrics in enumerate([['PM2.5', 'PM10'], ['NO2', 'SO2']], 1):
    fig, axes = plt.subplots(3, 2, figsize=(10, 9))
    fig.subplots_adjust(left=.08, right=.96, top=.79,
                        bottom=.13, hspace=.70, wspace=.23)
    title(fig, '10' + ('A' if part == 1 else 'B'),
          '多污染物分阶段变化矩阵',
          '2013=100 | 蓝色下降、金色上升、深色端点 | 各面板共用0-115刻度')
    for i, city in enumerate(cities):
        for j, metric in enumerate(metrics):
            ax = axes[i, j]
            field = 'pm25' if metric == 'PM2.5' else metric
            vals = [float(h.loc[(h.city==city) &
                               (h.year==y), field].iloc[0]) for y in stages[:-1]]
            vals += [float(p.loc[(p.city==city) &
                               (p.year==2024), metric].iloc[0])]
            index = np.array(vals) / vals[0] * 100
            delta = np.diff(index)
            assert np.isclose(100 + delta.sum(), index[-1])
            ax.axhline(100, color='#A5B8C4', ls='--', lw=.8)
            ax.bar([0,4], [100,index[-1]], width=.52, color=INK)
```

程序5 雷达、云朵词云与矩阵瀑布 6/6

04_advanced.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
for k, change in enumerate(delta, 1):
    color = '#2878B5' if change <= 0 else '#D49A3B'
    ax.bar(k, abs(change), bottom=min(index[k-1],index[k]),
           width=.52, color=color)
    ax.plot([k-.74,k-.26], [index[k-1]]*2,
            color='#9CAFBF', lw=1)
    ax.text(k, max(index[k-1],index[k])+4,
            f'{change:+.1f}', ha='center', fontsize=11)
    records.append([city,metric,stages[k-1],stages[k],
                    vals[k-1],vals[k],change,index[k]])
ax.plot([3.26,3.74], [index[-1]]*2, color='#9CAFBF')
ax.text(4,index[-1]+4,f'{index[-1]:.1f}',
        ha='center',fontsize=11,weight='bold')
ax.text(0,104,'100',ha='center',fontsize=10)
ax.set_title(f'{city} · {metric} {vals[0]:g} → {vals[-1]:g} μg/m³',
            loc='left',fontsize=12,pad=20,weight='bold')
ax.text(.99,1.08,f'累计 {index[-1]-100:+.1f}%',
        transform=ax.transAxes,ha='right',fontsize=10,
        color='#237E81')
ax.set(ylim=(0,115),xlim=(-.5,4.5),yticks=[0,50,100])
ax.set_xticks(range(5),['2013','13-16','16-19','19-24','2024'],
            fontsize=10)
ax.grid(axis='y',color=GRID,lw=.6)
ax.set_axisbelow(True)
ax.tick_params(length=0)
for spine in ax.spines.values():
    spine.set_visible(False)
fig.text(.055,.075,
        '中间标签为指数点变化；阶段时长为3年、3年、5年，不代表年均变化速度。',
        fontsize=10,color=MUTED)
name = '10_waterfall.png' if part == 1 else '10b_waterfall.png'
save(fig,name,
        '来源：国家统计局[20][23][26]及2024年官方公报[5][10][15][16]；不用于因果归因。')
pd.DataFrame(records,columns=['city','pollutant','start_year','end_year',
    'start_value','end_value','change_index_points','end_index']).to_csv(
    ROOT / 'results/waterfall_values.csv',index=False)
print('Figures 8, 9, 10A, 10B generated.')
```

程序6 省级季节哑铃图 1/2

05_seasonal.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
"""省级六污染物采暖期哑铃图；不将省级数据当作哈尔滨数据。"""
from pathlib import Path
import os
import numpy as np
import pandas as pd
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
from matplotlib import font_manager

ROOT = Path(__file__).resolve().parent
font_path = os.getenv('CHINESE_FONT')
if font_path:
    font_manager.fontManager.addfont(font_path)
    font = font_manager.FontProperties(fname=font_path).get_name()
else:
    names = {f.name for f in font_manager.fontManager.ttflist}
    font = next((n for n in ['Microsoft YaHei', 'SimHei',
                             'Noto Sans CJK SC', 'Droid Sans Fallback'] if n in names), None)
    if font is None:
        raise RuntimeError('需要中文字体或 CHINESE_FONT 环境变量')
plt.rcParams.update({'font.family': [font, 'DejaVu Sans'],
                    'axes.unicode_minus': False, 'text.color': '#16324B'})
s = pd.read_csv(ROOT / 'data/seasonal.csv')
metrics = ['PM2.5', 'PM10', 'NO2', 'SO2', 'CO-P95', 'O3-P90']
p = s.pivot(index='metric', columns='period', values='value').loc[metrics]
ratio = p['采暖期'] / p['非采暖期'] * 100
fig = plt.figure(figsize=(10, 6.5), facecolor='white')
fig.text(.055, .95, '图 11 采暖期与非采暖期的六污染物对比',
        fontsize=18, weight='bold', va='top')
fig.text(.055, .88, '黑龙江省 13 城市 · 2024 年 | 非采暖期浓度=100',
        fontsize=12, color='#637788')
ax = fig.add_axes([.36, .22, .56, .54])
ax.axvline(100, color='#A9BAC6', lw=1, ls='--')
for i, (m, v) in enumerate(ratio.items()):
```

程序6 省级季节哑铃图 2/2

05_seasonal.py 按页顺序拼接即为完整文件；不要把页眉加入代码。

```
ax.plot([100,v],[i,i],color='#AEC6D0',lw=5,
        solid_capstyle='round',zorder=1)
ax.scatter(100,i,s=90,color='#2878B5',zorder=2)
ax.scatter(v,i,s=90,color='#C28A39',zorder=3)
ax.text(v+(5 if v>=100 else -5),i,f'{v:.1f}',
        ha='left' if v>=100 else 'right',va='center',fontsize=12)
ax.set(ylim=(5.6,-.6),xlim=(50,240),yticks=[],xticks=[50,100,150,200])
ax.set_xlabel('浓度指数 (非采暖期=100)',fontsize=11,labelpad=10)
ax.grid(axis='x',color='#E7EDF2',lw=.7)
ax.tick_params(length=0)
for spine in ax.spines.values():
    spine.set_visible(False)
fig.text(.055,.79,'指标 / 原始浓度',fontsize=11,color='#637788')
fig.text(.24,.79,'采暖 / 非采暖',fontsize=10,color='#637788')
for i,m in enumerate(metrics):
    y = .22 + .54*(5.6-i)/6.2
    unit = 'mg/m³' if m=='CO-P95' else 'µg/m³'
    fig.text(.055,y,m,fontsize=12,va='center')
    fig.text(.185,y,unit,fontsize=9,color='#637788',va='center')
    fig.text(.265,y,f'{p.loc[m,"采暖期"]:g} / '
            f'{p.loc[m,"非采暖期"]:g}',fontsize=12,va='center')
fig.text(.055,.125,'● 非采暖期',fontsize=11,color='#2878B5')
fig.text(.25,.125,'● 采暖期',fontsize=11,color='#C28A39')
fig.text(.055,.075,'同期优良率: 采暖期90.3%, 非采暖期97.8%。'
        '省级背景不能替代城市观测; 差异不等于供暖的净因果效应。',
        fontsize=9,color='#637788')
fig.text(.055,.025,'来源: 黑龙江省生态环境厅[16], PDF 第19页。'
        'CO为日均P95, O3为日最大8小时P90; 其余为期间平均。',
        fontsize=8,color='#637788')
fig.savefig(R00T/'figures/11_seasonal.png',dpi=300)
fig.savefig(R00T/'figures/11_seasonal.svg')
pd.DataFrame({'metric':metrics,'heating_index':ratio.values}).to_csv(
    R00T/'results/seasonal_index.csv',index=False)
plt.close(fig)
```

5.2 AI 辅助提示词与核对说明

以下围绕选题、资料检索、数据核验、程序实现、图表分析和交付检查，整理 8 条可复制的 AI 辅助复现提示词。AI 用于辅助形成思路与检查表达，数据取值以官方资料为依据，计算结果由 Python 程序核验。

选题与范围：请依据“美丽中国”Python 实践作业要求，以北京、上海、哈尔滨 2010—2024 年空气质量为案例，设计下载、清洗、统计、十类图表和报告路线。明确 2010—2012 年 PM2.5 原表缺失，不编造或插值。

官方资料查找：请核对国家统计局 2010—2019 年空气质量表和三市 2020—2024 年公报，记录机构、题名、链接、城市行及单位。早期浓度换算为 $\mu\text{g}/\text{m}^3$ ；2018 年上海达标天数核对为 295 天。

统计口径复核：区分新旧评价体系、城市与省级范围、年均与百分位浓度。上海 2024 年按原件保留 88.5%，六类等级天数合计 365；哈尔滨 2024 年分母不一致保留说明。不可凭推测添加沙尘扣除解释，不用省级值补城市缺失。

数据核验程序：请使用 Pandas 编写数据核验脚本，检查城市年份完整性、重复主键、缺失值、数值类型、比例范围及来源编号。异常时直接报错，不用随机数补齐，不默默删除记录。

图表生成：保留原有图表，词云改为云朵掩膜并使用 13 份官方节选的真实计数；瀑布扩展为三市四指标三阶段的 12 面板矩阵，统一 2013=100；新增省级采暖期哑铃图。保留原始值、单位、来源与解释边界，不用装饰代替分析。将新增 CO、O3 纳入图 2 六面板和图 8 六轴雷达，等级天数纳入图 6；词云计数条使用圆角且保持线性长度。

分析计算：请计算三市 2013—2024 年 PM2.5 端点降幅、均值、中位数与极差，并保留近五年的专题比较。相关系数每市仅 5 点，说明哈尔滨优良率限制；早期缺失不得补零，不能跨旧新评价比例计算同比。

逐图撰写：请逐图写出主要数值、可比较的差异、不能得出的结论和实践启示。说明上海 2022 年 PM2.5 下降而 AQI 优良率下降的现象，并用官方污染日构成辅助解释。

交付核对：检查 10 种图形、12 张图、六个程序完整源码和至少 5 条提示词。核对 45 条年度记录、36 个 PM2.5 有效值与 55 条扩展专题记录；核验瀑布加和、词频计数、等级天数和单位。检查封面、目录、分页与分工一致性。

复核原则：AI 回答不作为独立数据来源。原始数据、来源索引、处理步骤与计算结果应能相互核对，分析结论应与数据范围和统计口径保持一致。

参考文献

图表[1]—[29]与下列编号对应。历史年度与年鉴出版年份不同；数据以政府官网原件为依据。新增历史记录固定采用国家统计局公布值，下载原件、网址、检索日期与文件摘要见项目 raw 目录。

[1] 北京市生态环境局. 2020 年 12 月北京市环境空气质量状况[EB/OL]. 2021[2026-10-15].

<https://sthjj.beijing.gov.cn/bjhrb/index/xxgk69/sthjlyzgw/1718880/1718889/10922039/index.html>

数据定位：一、全市空气质量总体状况，1—12 月数据。

[2] 北京市生态环境局. 2021 年北京市生态环境状况公报解读[EB/OL]. 2022[2026-10-15].

<https://sthjj.beijing.gov.cn/bjhrb/index/xxgk69/zfxgk43/fdzdgknr2/zcfb/zcjd89/325832911/index.html>

数据定位：空气质量首次全部达标段落。

[3] 北京市生态环境局. 2022 年北京市空气质量新闻发布会全文实录[EB/OL]. 2023[2026-10-15].

<https://sthjj.beijing.gov.cn/bjhrb/index/xxgk69/zfxgk43/fdzdgknr2/ywdt28/xwfb/326013538/index.html>

数据定位：刘保献介绍 2022 年全年空气质量段落。

[4] 北京市生态环境局. 2023 年 12 月北京市环境空气质量状况[EB/OL]. 2024[2026-10-15].

<https://sthjj.beijing.gov.cn/bjhrb/index/xxgk69/sthjlyzgw/1718880/1718889/436346487/index.html>

数据定位：一、全市空气质量总体状况，1—12 月数据。

[5] 北京市生态环境局. 2024 年北京市收获 290 个优良天 为历年最多[EB/OL]. 2025[2026-10-15].

<https://sthjj.beijing.gov.cn/bjhrb/index/xxgk69/zfxgk43/fdzdgknr2/ywdt28/xwfb/543502506/index.html>

数据定位：主要污染物年均浓度和优良天占比段落。

参考文献 续

[6] 上海市生态环境局. 2020 上海市生态环境状况公报[EB/OL]. 2021[2026-10-15].

<https://sthj.sh.gov.cn/hbzhywpt1143/hbzhywpt1144/20210607/7550cad81da8424eadb07a2c96c1208c.html>

数据定位：一、全市生态环境质量明显改善。

[7] 上海市生态环境局. 上海市生态环境局发布 2021 年度生态环境状况公报[EB/OL]. 2022[2026-10-15].

<https://sthj.sh.gov.cn/hbzhywpt1272/hbzhywpt1157/20220722/40469039c5ac4f8aa9c23576324607b2.html>

数据定位：一、全市生态环境质量持续改善。

[8] 上海市生态环境局. 2022 上海市生态环境状况公报[EB/OL]. 2023[2026-10-15].

<https://sthj.sh.gov.cn/cmsres/43/431903995fad4918af80e2d021b431a8/f289185c2e7e4f8ff6b6aa6376309821.pdf>

数据定位：PDF 第 5 页空气质量等级和污染日构成；第 5—7 页六项污染物、酸雨与道路扬尘。

[9] 上海市生态环境局. 2023 上海市生态环境状况公报[EB/OL]. 2024[2026-10-15].

<https://sthj.sh.gov.cn/hbzhywpt1143/hbzhywpt1144/20240605/3aa7abac13f8440ba5eff23466e6f82d.html>

数据定位：正文 2023 年 AQI 优良率和 PM_{2.5} 年均浓度。

[10] 上海市生态环境局. 2024 上海市生态环境状况公报[EB/OL]. 2025[2026-10-15].

<https://sthj.sh.gov.cn/cmsres/5a/5a91d13120d9455c920568ed8190a017/>

<feb4a53fa6fa665aa655fc9971f3d616.pdf>

数据定位：PDF 第 5 页右栏优良率 88.5%、各等级天数；第 6—7 页六项污染物、酸雨与道路扬尘。

参考文献 续

[11] 哈尔滨市统计局、国家统计局哈尔滨调查队. 2020 年哈尔滨市国民经济和社会发展统计公报[EB/OL]. 2021[2026-10-15].

https://www.harbin.gov.cn/haerbin/c104569/202104/c01_70393.shtml

数据定位：十二、资源环境和安全生产，环境段落。

[12] 哈尔滨市统计局、国家统计局哈尔滨调查队. 2021 年哈尔滨市国民经济和社会发展统计公报[EB/OL]. 2022[2026-10-15].

https://www.harbin.gov.cn/haerbin/c104569/202205/c01_70402.shtml

数据定位：十二、能源、资源环境和安全生产，环境段落。

[13] 哈尔滨市统计局、国家统计局哈尔滨调查队. 2022 年哈尔滨市国民经济和社会发展统计公报[EB/OL]. 2023[2026-10-15].

https://www.harbin.gov.cn/haerbin/c104569/202305/c01_730626.shtml

数据定位：公报印刷第 22 页（PDF 第 23 页），PM_{2.5} 为 38；采用本年公报原值。

[14] 哈尔滨市统计局、国家统计局哈尔滨调查队. 2023 年哈尔滨市国民经济和社会发展统计公报[EB/OL]. 2024[2026-10-15].

https://www.harbin.gov.cn/haerbin/c104569/202407/c01_997199.shtml

数据定位：公报印刷第 18 页（PDF 第 19 页），PM_{2.5} 为 36，优良率 83.3%。

[15] 哈尔滨市统计局、国家统计局哈尔滨调查队. 2024 年哈尔滨市国民经济和社会发展统计公报[EB/OL]. 2025[2026-10-15].

https://www.harbin.gov.cn/haerbin/c104569/202506/c01_1063449.shtml

数据定位：公报印刷第 18 页（PDF 第 19 页）；原文 365 天、312 天、85.2%存在分母不一致，比例保留公布值。

[16] 黑龙江省生态环境厅. 2024 年黑龙江省生态环境质量状况[EB/OL]. 2025[2026-10-16].

[https://www.hlj.gov.cn/hlj/c108433/202502/31812306/files/2024 年黑龙江生态环境质量状况.pdf](https://www.hlj.gov.cn/hlj/c108433/202502/31812306/files/2024%20黑龙江生态环境质量状况.pdf)

数据定位：印刷第 19 页（PDF 第 21 页）PM₁₀ 排名续表，哈尔滨 2024 年 62 $\mu\text{g}/\text{m}^3$ ；第 15 页优良率 85.2%与市统计公报相符。

参考文献 历史数据与口径说明

[17] 国家统计局. 主要城市空气质量指标 (2010 年) [EB/OL]. 2023[2026-10-20].

https://www.stats.gov.cn/zt_18555/ztsj/hjtjzl/2010/202303/t20230302_1921714.html

数据定位：国家统计局环境统计资料 2010 年栏目，2023 年网站迁移页面；北京、上海、哈尔滨行。

[18] 国家统计局. 主要城市空气质量指标 (2011 年) [EB/OL]. 2023[2026-10-20].

https://www.stats.gov.cn/zt_18555/ztsj/hjtjzl/2011/202303/t20230302_1921484.html

数据定位：国家统计局环境统计资料 2011 年栏目，2023 年网站迁移页面；北京、上海、哈尔滨行。

[19] 国家统计局. 主要城市空气质量指标 (2012 年) [EB/OL]. 2023[2026-10-20].

https://www.stats.gov.cn/zt_18555/ztsj/hjtjzl/2012/202303/t20230302_1921248.html

数据定位：国家统计局环境统计资料 2012 年栏目，2023 年网站迁移页面；北京、上海、哈尔滨行。

[20] 国家统计局. 主要城市空气质量指标 (2013 年) [EB/OL]. 2023[2026-10-20].

https://www.stats.gov.cn/zt_18555/ztsj/hjtjzl/2013/202303/t20230302_1921012.html

数据定位：国家统计局环境统计资料 2013 年栏目，2023 年网站迁移页面；北京、上海、哈尔滨行。

参考文献 历史数据与口径说明

[21] 国家统计局. 主要城市空气质量指标 (2014 年) [EB/OL]. 2023[2026-10-20].

https://www.stats.gov.cn/zt_18555/ztsj/hjtjzl/2014/202303/t20230303_1924030.html

数据定位：国家统计局环境统计资料 2014 年栏目，2023 年网站迁移页面；北京、上海、哈尔滨行。

[22] 国家统计局. 2015 年环保重点城市空气质量情况[EB/OL]. 2016[2026-10-20].

<https://www.stats.gov.cn/sj/ndsjsj/2016/html/0819CH.jpg>

数据定位：中国统计年鉴 2016 空气质量表；北京、上海、哈尔滨行。

[23] 国家统计局. 2016 年环保重点城市空气质量情况[EB/OL]. 2017[2026-10-20].

<https://www.stats.gov.cn/sj/ndsjsj/2017/html/CH0819.jpg>

数据定位：中国统计年鉴 2017 空气质量表；北京、上海、哈尔滨行。

[24] 国家统计局. 2017 年环保重点城市空气质量情况[EB/OL]. 2018[2026-10-20].

<https://www.stats.gov.cn/sj/ndsjsj/2018/html/CH0819.jpg>

数据定位：中国统计年鉴 2018 空气质量表；北京、上海、哈尔滨行。

参考文献 历史数据与口径说明

[25] 国家统计局. 2018 年环保重点城市空气质量情况[EB/OL]. 2019[2026-10-20].

<https://www.stats.gov.cn/sj/ndsjs/2019/html/C0817.jpg>

数据定位：中国统计年鉴 2019 空气质量表；北京、上海、哈尔滨行。

[26] 国家统计局. 2019 年主要城市空气质量情况[EB/OL]. 2020[2026-10-20].

<https://www.stats.gov.cn/sj/ndsjs/2020/html/C0817.jpg>

数据定位：中国统计年鉴 2020 空气质量表；北京、上海、哈尔滨行。

[27] 环境保护部. 环境空气质量不容乐观 大气污染治理任重道远 [EB/OL]. 2014[2026-10-20].

https://www.mee.gov.cn/gkml/sthjbgw/qt/201403/t20140325_269651.htm

数据定位：2013 年实施新标准的 74 个城市，PM2.5、CO 和 O3 纳入评价指标。

预览版仅供参考请以最终提交版为准

参考文献 扩展数据与统计资料

[28] 北京市生态环境局. 2024 Report on the State of the Ecology and Environment in Beijing[EB/OL]. 2025[2026-10-22].

<https://sthjj.beijing.gov.cn/bjhrb/index/xxgk69/sthjlyzwg/1718880/1718881/1718882/743615203/2025121215323323447.pdf>

数据定位：英文版 PDF 第 4 页 CO、O₃；第 13 页声环境。

[29] 生态环境部. 生态环境部发布《2024 中国生态环境状况公报》[EB/OL]. 2025[2026-10-22].

https://www.mee.gov.cn/ywgz/sthjcgj/hjzljcypj/202506/t20250605_1120773.shtml

数据定位：一、空气质量稳中向好；正文用于扩展生态环境语料。

预览版仅供参考请以最终提交版为准