Методы визуализации и понижения размерности

Формально задача снижения размерности для эмпирической модели с одним откликом состоит в том, чтобы найти с приемлимой точностью преобразование матрицы предикторов X размерности $p \times n$ в матрицу X^{dr} размерности $p \times 2$ или покрайней мере $p \times 3$,чтобы обеспечить возможность эффективного визуальногог анализа исходных данных:

$$\Psi^{dr} : X_{p \times n} \rightarrow X^{dr}_{p \times m} \, (1)$$

где $n >> m (m = 2, 3)$;dr обозначает снижение размерности (dimension reduction).

Отображение (алгоритм) (1) каждую строку исходной матрицы размерности n преобразует в строку новой матрицы размерности m в соответствии с определенными правилами,которые должны в максимальной степени сохранять структуру и закономерности в исходных данных. При таком преобразовании каждая строка новой матрицы может изображаться точкой в двухмерном или трехмерном пространстве, что допускает возможности эффективного визуального анализа исходных данных, но чтобы этот анализ был адекватным исходным данным алгоритм прехода к малой размерности должет отвечать определенным требованиям:

- Каждый объект с большой размерностью представляется объектом с малой размерностью - Сохранение “соседства” двух объектов в разных пространствах - Удаленные точки соответствуют отличающимся объектам - Масштабируемость

Существует достаточно много методов понижения размерности исходного пространства предикторов расмотрим наиболее известные из них.

Метод главных компонент PCA

Метод главных компонент (англ. $PrincipalComponentsAnalysis, PCA$) — один из основных способов уменьшить размерность данных, потеряв наименьшее количество информации. Изобретен К. Пирсоном (англ. $KarlPearson$) в 1901 г. Применяется во многих областях, таких как распознавание образов, компьютерное зрение, сжатие данных и т.п. Вычисление главных компонент сводится к вычислению собственных векторов и собственных значений ковариационной матрицы исходных данных или к сингулярному разложению матрицы данных. Иногда метод главных компонент называют преобразованием Карунена-Лоэва (англ. Karhunen-Loeve) или преобразованием Хотеллинга (англ. Hotelling transform).

Формальная постановка задачи

В качестве исходных данных PCA используется матрица предикторов, которая имеет следующий вид:

$$X = \begin{pmatrix} x_{1,1} & x_{1,2} \ldots & x_{1,\mathrm{n}} \\ x_{2,1} & x_{2,2} \ldots & x_{2,\mathrm{n}} \\ x_{p,1} & x_{p,2} \ldots & x_{p,n} \end{pmatrix} (2)$$

В этой матрице $\forall\, x_{i,j}$ представляет собой действительное число, полученное в результате измерения соответстующего свойства у некотого объекта в метрической шкале (шкале отношений). Ведем следующие математически абстракции. Считаем, что каждый столбец матрицы предикторов представляет собой соответсвующий признак, задаваемый в виде значений некоторой функции $f_j(x), j = \overline{1,n}$. Тогда объекты будем отождествлять с их признаковыми описаниями:$X_i \equiv (\, f_1(\, X_i\,), ..., f_n(\, X_i\,)\,), i = \overline{1,n}$. Следовательно имеем матрицу, строки которой соответствуют признаковым описаниям объектов в пространстве $X = \Re^n$:

$$F_{p\times n} = \begin{pmatrix} f_1(\, X_1\,) & \ldots & f_n(\, X_1\,) \\ \ldots & \ldots & \ldots \\ f_1(\, X_p\,) & & f_n(\, X_p\,) \end{pmatrix} = \begin{pmatrix} X_1 \\ \ldots \\ X_p \end{pmatrix} (3)$$

Обозначим через $X_i^{dr} = (\, g_1(X_i), ..., g_m(X_i)\,)$ признаковые описания тех же объектов в новом пространстве $X^{dr} = \Re^m$ меньшей размерности,$m < n$:

$$G_{p\times m} = \begin{pmatrix} g_1(X_1) & \ldots & g_m(X_1) \\ \ldots & \ldots & \ldots \\ g_1(X_p) & & g_m(X_p) \end{pmatrix} = \begin{pmatrix} X_1^{dr} \\ \ldots \\ X_p^{dr} \end{pmatrix} (4)$$

Потребуем, чтобы исходные признаковые описания можно было восстановить по новым описаниям с помощью некоторого линейного преобразования, определяемого матрицей $U = (u_{js})_{n\times m}$:

$$f_j^*(X) = \sum_{s=1}^{m} g_s(X) \cdot u_{js}, j = \overline{1,n} (5)$$

или в векторной записи: $X^* = X^{dr}\, U^T$. Восстановленное описание X^* не обязано в точности совпадать с исходным описанием X, но их отличие на объектах исходной выборки должно быть как можно меньше при выбранной размерности m. Будем искать одновременно и матрицу

новых признаковых описаний G, и матрицу линейного преобразования U, при которых суммарная невязка $\Delta^2(G,U)$ восстановленных описаний минимальна:

$$\Delta^2(G,U) = \sum_{i=1}^{p} \|X_i^* - X_i\|^2 = \sum_{i=1}^{p} \left\|X_i^{dr} \cdot U^T - X_i\right\|^2 = \left\|G \cdot U^T - F\right\|^2 \to \min_{GU} \quad (6)$$

Будем предполагать, что матрицы G и U невырождены: $rankG = rankU = m$. Иначе существовало бы представление $\overline{G} \cdot \overline{U^T} = G \cdot U^T$ с числом столбцов в матрице $\bar{G}$, меньшим m. Поэтому интересны лишь случаи, когда $m \leq rankF$.

Решение задачи

Решение сформулированной задачи даёт следующая теорема.

Если $m \leq rankF$, то минимум $\Delta^2(G,U)$ достигается, когда столбцы матрицы U есть собственные векторы $F^T F$, соответствующие m максимальным собственным значениям. При этом $G = FU$, матрицы U и G ортогональны.

Доказательство:

Запишем необходимые условия минимума:

$$\begin{cases} \frac{\partial \Delta^2}{\partial G} = (G\,U^T - F)\,U = 0 \\ \frac{\partial \Delta^2}{\partial U} = G^T(G\,U^T - F) = 0 \end{cases} \quad (7)$$

Поскольку искомые матрицы G и U невырождены, отсюда следует:

$$\begin{cases} G = F\,U(U^T\,U)^{-1} \\ U = F^T\,G\,(\,G^T\,G\,)^{-1} \end{cases} \quad (8)$$

Функционал $\Delta^2(G,U)$ зависит только от произведения матриц GU^T, поэтому решение задачи $\Delta^2(G\,U) \to \min\limits_{G,U}$ определено с точностью до произвольного невырожденного преобразования R : $G\,U^T = G\,R(R^{-1}\,U^T)$. Распорядимся свободой выбора R так, чтобы матрицы U^TU и G^TG оказались диагональными.

Покажем, что это всегда возможно.

Пусть $\overset{*}{G}\,\overset{*}{U}^T$ - произвольное решение.

Матрица $\overset{*}{U}^T\,\overset{*}{U}$ симметричная, невырожденная, положительно определенная, поэтому существует невырожденная матрица $S_{m?m}$ такая, что

$S^{-1}\overset{*}{U}^{T}\overset{*}{U}(S^{-1})^{T} = I_m$.

Матрица $S^{T}\overset{*}{G}^{T}\overset{*}{G}S$ симметричная и невырожденная, поэтому существует ортогональная матрица $T_{m?m}$ такая, что

$T(^{T}S^{T}\overset{*}{G}^{T}\overset{*}{G}S)T = diag(\lambda_1, ..., \lambda_m) \equiv \Lambda$ — диагональная матрица. По определению ортогональности $T^T T = I_m$.

Преобразование $R = ST$ невырождено. Положим $G = \overset{*}{G}RU^{T} = R^{-1}\overset{*}{U}^{T}$. Тогда

$$G^T G = T(^{T}S^{T}\overset{*}{G}^{T}\overset{*}{G}S)T = \Lambda$$

$$U^T U = T^{-1}(S^{-1}\overset{*}{U}^{T}\overset{*}{U}(S^{-1})^{T}(T^{-1})^{T} = (T^T T)^{-1} = I_m$$

В силу $GU^{T} = \overset{*}{G}\overset{*}{U}^{T}$ матрицы G и U являются решением задачи $\Delta^2(GU) \to \min\limits_{G,U}$ И удовлетворяют необходимому условию минимума. Подставим матрицы G и U в (8)

$$\begin{cases} G = FU(U^T U)^{-1} \\ U = F^T G(G^T G)^{-1} \end{cases} \quad (9)$$

Благодаря диагональности G^TG и U^TU соотношения (9) существенно упростятся:

$$\begin{cases} G = FU \\ U\Lambda = F^T G \end{cases} \quad (10)$$

Подставим первое соотношение системы (10) во второе, получим $U\Lambda = F^T FU$. Это означает, что столбцы матрицы U обязаны быть собственными векторами матрицы F^TF, а диагональные элементы $\lambda_1, ..., \lambda_m$ - соответствующими им собственными значениями.

Аналогично, подставив второе соотношение в первое, получим $G\Lambda = FF^TG$, то есть столбцы матрицы G являются собственными векторами FF^T, соответствующими тем же самым собственным значениям.

Подставляя G и U в функционал $\Delta^2(G, U)$, находим:

$$\Delta^2(G, U) = \left\|F - GU^T\right\|^2 = tr(F^T - UG^T)(F - GU^T) =$$

$$tr\, F^T\,(\,F - G\,U^T\,) = tr\, F^T\, F - tr\, F^T\, G\, U^T\,) = \|F\|^2 - tr\, U\, \Lambda\, U^T =$$

$$\|F\|^2 - tr\, \Lambda = \sum\nolimits_{j=1}^{n} \lambda_j - \sum\nolimits_{j=1}^{m} \lambda_j - \sum\nolimits_{j=m+1}^{n} \lambda_j,$$

где $\lambda_1, ..., \lambda_n$ - все собственные значения матрицы $F^T F$. Минимум Δ^2 достигается, когда $\lambda_1, ..., \lambda_m$ — наибольшие m из n собственных значений. Собственные векторы $u_1, ..., u_m$, отвечающие максимальным собственным значениям, называют главными компонентами.

Свойства

Связь с сингулярным разложением

Если $m = n$, то $\Delta^2\,(\,G\,U\,) = 0$. В этом случае представление $F = GU^T$ является точным и совпадает с сингулярным разложением: $F = GU^T = VDU^T$, если положить $G = VD$ и $\Delta = D^2$. При этом матрица V ортогональна: $V^T V = I_m$.

Если $m < n$, то представление $F \approx G\,U^T$ является приближённым. Сингулярное разложение матрицы GU^T получается из сингулярного разложения матрицы F путём отбрасывания (обнуления) $n?m$ минимальных собственных значений.

Преобразование Карунена–Лоэва

Диагональность матрицы $G^T G = \Lambda$ означает, что новые признаки $g_1, ..., g_m$ не коррелируют на объектах выборки. Ортогональное преобразование U называют декоррелирующим или преобразованием Карунена–Лоэва. Если $m = n$, то прямое и обратное преобразование вычисляются с помощью одной и той же матрицы $U : F = GU^T$ и $G = FU$.

Эффективная размерность

Главные компоненты содержат основную информацию о матрице F. Число главных компонент m называют также эффективной размерностью задачи. На практике её определяют следующим образом. Все собственные значения матрицы $F^T F$ упорядочиваются по убыванию: $\lambda_1 \geq ... \geq \lambda_n \geq 0$. Задаётся пороговое значение $\epsilon \in [0, 1]$, достаточно близкое к нулю, и определяется наименьшее целое m, при котором относительная погрешность приближения матрицы F не превышает ϵ:

$$E(m) = \frac{\left\|G\,U^T - F\right\|^2}{\|F\|^2} = \frac{\lambda_{m+1} + ... + \lambda_n}{\lambda_1 + ... + \lambda_n} \leq \varepsilon$$

Величина $E(m)$ показывает, какая доля информации теряется при замене исходных признаковых описаний длины n на более короткие описания длины m. Метод главных компонент особенно эффективен в тех случаях, когда $E(m)$ оказывается малым уже при малых значениях m. Если задать число ϵ из априорных соображений не представляется возможным, прибегают к критерию «крутого обрыва». На графике $E(m)$ отмечается то значение m, при котором происходит резкий скачок: $E(m-1) >> E(m)$, при условии, что $E(m)$ уже достаточно мало.

Визуализация многомерных данных

Метод главных компонент часто используется для представления многомерной выборки данных на двумерном графике. Для этого полагают $m = 2$ и полученные пары значений $(g_1(x_i), g_2(x_i)), i = 1, ..., p$ наносят как точки на график. Проекция на главные компоненты является наименее искаженной из всех линейных проекций многомерной выборки на какую-либо пару осей. Как правило, в осях главных компонент удаётся увидеть наиболее существенные особенности исходных данных, даже несмотря на неизбежные искажения. В частности, можно судить о наличии кластерных структур и выбросов. Две оси g_1 и g_2 отражают «две основные тенденции» в данных. Иногда их удаётся интерпретировать, если внимательно изучить, какие точки на графике являются «самыми левыми», «самыми правыми», «самыми верхними» и «самыми нижними». Этот вид анализа не позволяет делать точные количественные выводы и обычно используется с целью понимания данных.

Пределы применимости и ограничения эффективности метода

Метод главных компонент применим всегда. Распространённое утверждение о том, что он применим только к нормально распределённым данным (или для распределений, близких к нормальным) неверно: в исходной формулировке К. Пирсона ставится задача об аппроксимации конечного множества данных и отсутствует даже гипотеза о их статистическом порождении, не говоря уж о распределении. Однако метод не всегда эффективно снижает размерность при заданных ограничениях на точность $E(m)$. Прямые и плоскости не всегда обеспечивают хорошую аппроксимацию. Например, данные могут с хорошей точностью следовать какой-нибудь кривой, а эта кривая может быть сложно расположена в пространстве данных. В этом случае метод главных компонент для приемлемой точности потребует нескольких компонент (вместо одной), или вообще не даст снижения размерности при приемлемой точности.

Большие неприятностей могут доставить данные сложной топологии.

Originalsource

Статистический подход к обоснованию и интерпретации PCA

Выше нами рассматривался подход к обоснованию метода PCA, который основывался на постулатах и свойствах, связанных с линейными преобразованиями исходной матрицы предикторов X. Сейчас рассмотрим статистический подход, базирующися на представлении,что мы имеем дело с выборкой из генеральной совокупности системы случайных переменных (предикторов). При этом предполагается, что наблюдаемые переменные являются линейной комбинацией некоторых латентных факторов. Одни из этих факторв являются общими для нескольких переменных (предикторов), а другие характерны для каждого исходного предиктора. Последние факторы-ортогональны друг другу.Следовательно, они не вносят вклад в ковариацию между переменными (предикторами). То есть,только общие факторы, число которых предполагается гораздо меньшим числа наблюдаемых переменных, вносят вклад в ковариацию между ними.

При использовании данных идей и свойст линейных преоразований можно точно идентифицировать латентную факторную структуру путем исследования результирующей ковариационной матрицы, если она не является слишком сложной. Но так как на исследуемую матрицу оказывают влияние различные случайные и неслучайные ошибки и в результате она будет отличаться от корреляционной матрицы, обусловленной генеральной совокупностью.Следовательно, на практике невозможно получить точную структуру факторной модели, можно только пытаться найти оценки параметров факторной стрктуры, с использованием определенных статистических критериев. Решение задачи выделения общих латентных факторов предполагает реализацию следующих этапов:

- подготовка соотвествующей ковариационной матрицы;
- выделение первоначальны (ортогональных) факторов;
- вращение с целью получения окончательного решения.

Остановимся на этапе выделения первоначальных факторов, где может использоваться модель общих факторов, а также анализ главных компонент. Оба метода являются эффективными способами исследования взаимосвязей между переменными. Основное отлтчие между ними заключается в том, что главные компоненты являются линейными функциями от наблюдаемых переменных, а общие факторы не выражаются через комбинацию наблюдаемых переменных.

Основная цель выделения первичных факторов заключается в определении минимального числа общих факторов, которые удовлетворительно воспроизводят корреляции между наблюдаемыми переменными. Если бы наблюдаемые переменные представляли генеральную совокупность и не были подвержены ошибкам измерения, то для соотвествующей этим условиям редуцированной корреляционная матрицы существует точное соотвествие между минимальным числом общих факторов и рангом этой матрицы (в редуцированной корреляционной матрице общности помещаются на главную диагональ). Но если выборка является случайной, то возникает задача найти критерий, с помощью которого можно было бы оценить минимально необходимое число общих факторов. Эта задача сводится к определению правила остановки при выделении общих фаторов. Т о есть нужно определить момент, когда расхождение меду вычисленными и наблюдаемыми корреляциями может быть приписано случайности выборки.

Основная стратегия решения этой задачи включает проверку гипотезы о минимальном числе общих факторов, необходимых для воспроизведения наблюдаемых корреляций. Обычно начинают с однофакторной модели(достаточно одного фактора). Данная "гипотеза"проверяется с помощью критерия. .Если она не подтверждается и расхождение межу вычесленной и наблюдаемой корреляционными матрицами статистически значимо, то оценивается модель с еще одним фактором. Это т процесс продожается дотех пор пока расхождение может приписано только случайности выборки. Хотя принцип этой основной стратегии прост, его применение-разнообразно, поскольку есть разные критерии соотвествия(или минимальной невязки).

Всвязи с этим вделяются следующие методы:

- Главные компоненты PCA;
- Невзвешанный МНК;
- Обобщенный МНК;
- Максимум правдоподобия;
- Альфа факторизация;
- Анализ образов.

В нашей работе мы остановимся методе главных компонент.

Главные компоненты, собственные значения и вектора.

Анализ главных компонент - это метод преобразования последовательности наблюдаемых переменных(в нашей терминологии предикторов модели) в другую последовательность переменных (новых предик-

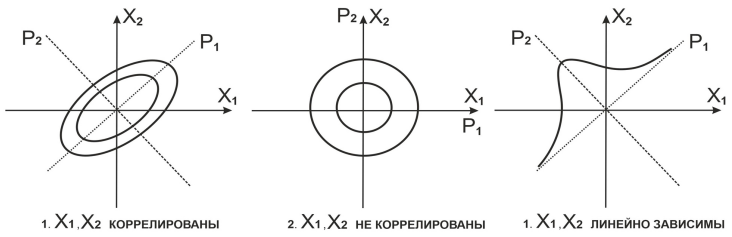


Рис. 1: Главные оси двумерных распределений

торов). Наиболее простой способ пояснить внутреннюю логику метода сводится к его изучению в двумерном случае. Предположим, что есть две случайные переменные X_1 и X_2 с совместным нормальным распределением. Возможны следующие три случая (рис.1).

Совместное нормальное распределение велечин , имеющих совместную положительную корреляцию на рис под номером 1 с помощьюкривых равных вероятностей. Эти кривые показывают, что благодаря положжтельной корреляционной связи между X_1 и X_2 данные представляют собой кластер, котором боьшие величины X_1 имеют тенденцию соотвествовать большим значеним X_2 (и наоборот). Таким образом, в большинстве случаев точки попадают в 1 и 3 квадранты и реже в 2 и 4. Кривые равных имеют форму элипсов (в общем случае элипсоидов), две оси которых изображены пунктирными линиями. Главная ось P_1 проходит по линии,в доль которой расположена основная часть данных; вторая ось P_2 - по линии, вдоль которой расположена меньшая часть данных.

Теперь предположим, что нужно представить точкив терминах только одной размерности (оси). В этом случае естественно выбрать ось P_1, потому чтов целом она ближе описывает данные наблюдений.Тогда первая компонента есть представление точек на оси P_1. Если мы описываем каждую точкув терминах P_1 и P_2, то потери информации не произойдет. Но можно сказать, что первая ось (первая компонента) является наиболее информативной в описании точек, так как свзь меду X_1 и X_2 становиться сильнее. В том случае когда переменные X_1 и X_2 связаны линейно, первая главная компонента будет содержать всю информациюЮ необходимую для описания каждой точки. Если переменные X_1 и X_2

независимы, то главная ось отсутствует и следовательно метод главныз компонент не может обеспечить даже минимальное сокращение (сжатие) размерности наблюдений.

Понятие главных осей относиться не только к нормальным распределениям. В общем случае задается линией, для которой сумма квадратов расстояний до всех точек наблюдения минимальна. Сравнение метода главных компонентс принципом наименьших квадратов(МНК) может объяснить это определение. При нахождении линии регрессии $X_2 = a + bX_1$ МНК мы минимизируем сумму квадратов расстояний между X_2 и $\overline{X_2}$, где это расстояние измеряется по линии параллельной оси X_2 и перпендикулярной оси X_1. При нахождении главной оси мы минимизируем расстояние от точек до главной оси (т.е расстояние по перпендикуляру к главной оси). Это различие проиллюстрированона рисунке (рис. Сравнение регрессий, полученныхс помощью МНК и главных осей.)

Так как первая компонента определена таким образом, что основная доля информации содержится в ней (максимальная дисперсия в направлении этой компоненты), вторая компонента определяется аналогичным способом.Следовательно, в двумерном случае после фиксирования первой компоненты вторая известна автоматически. Если X_2 не является линейной функцией X_1, то главных компонент будет две, только в этом случае полностью можно описать их совместное распределение. Таким образом, если методом главных компонент достигается сжатие данных (выделение только нескольких первых компонент) это означает,чо они объясняют максимальную долю общей дисперсии данных.

При наличии более двух переменных принцип определения главных компонент тот же. Данные в трехмерном случае можно представить следующим образом (рис. 3)

Для трехмерного нормального распределения поверхность равной вероятности будет ограничивать элипсоид (рис.4)

В качестве первой главной компоненты нужно выбрать такую координату, что соответствующая координатная ось направлена вдоль того направления, вдоль которого разброс точек самый большой – то есть вдоль длинной оси эллипсоида (рис. 5).

Так мы ввели первую координату, PC1. Но нужно ввести ещё две. Как и в двумерном случае, эти координатные оси должны быть перпендикулярны PC1.

Но теперь этого недостаточно для их однозначного определения: на плоскости есть лишь одна прямая, перпендикулярная данной и прохо-

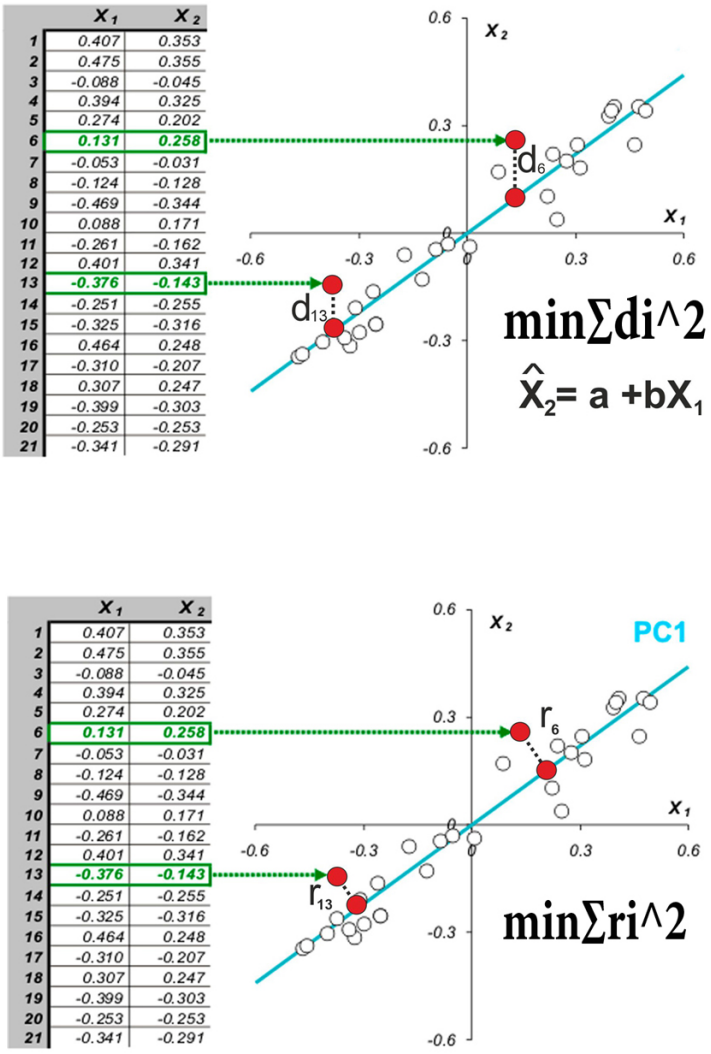


Рис. 2: МНК и метод главных компонент

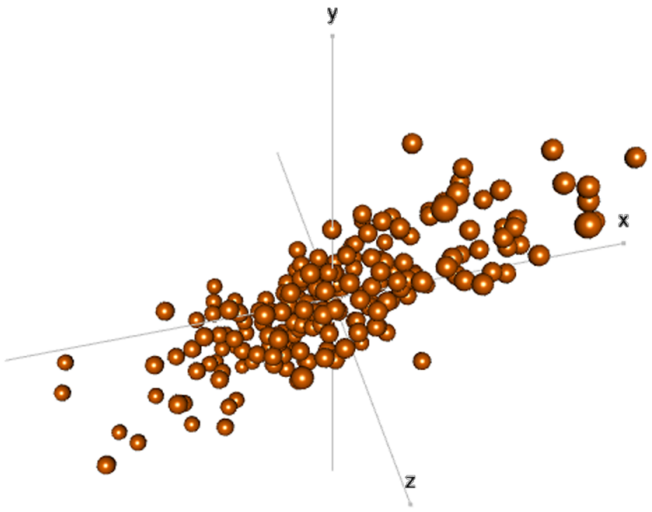


Рис. 3: Данные в трехмерном пространстве

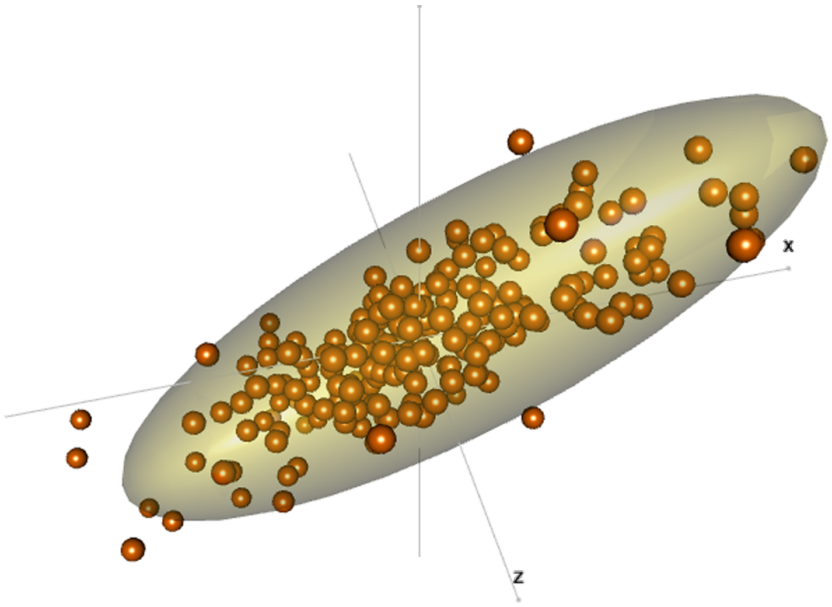


Рис. 4: Элипсоид - поверхность равной вероятности

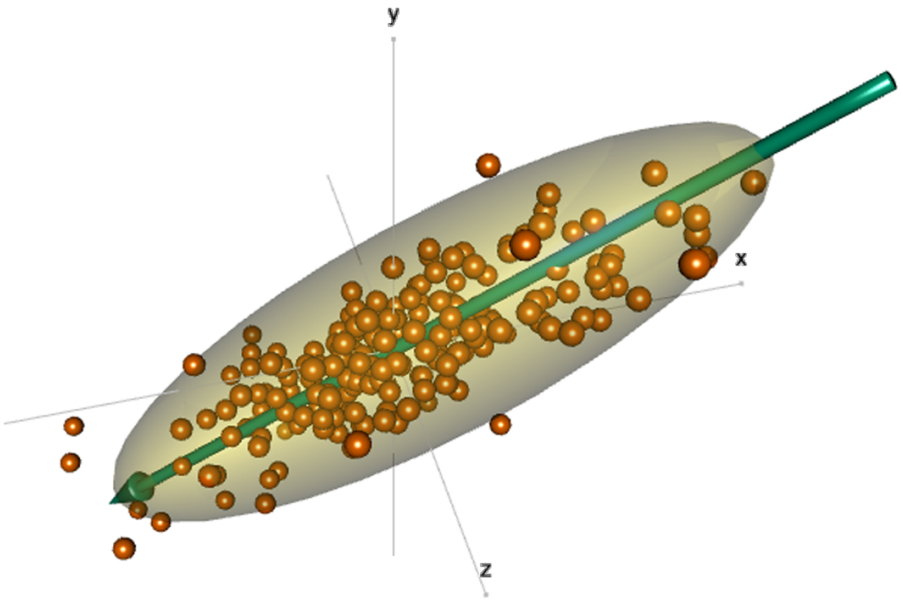


Рис. 5: Первая компонента в трехмерном пространстве

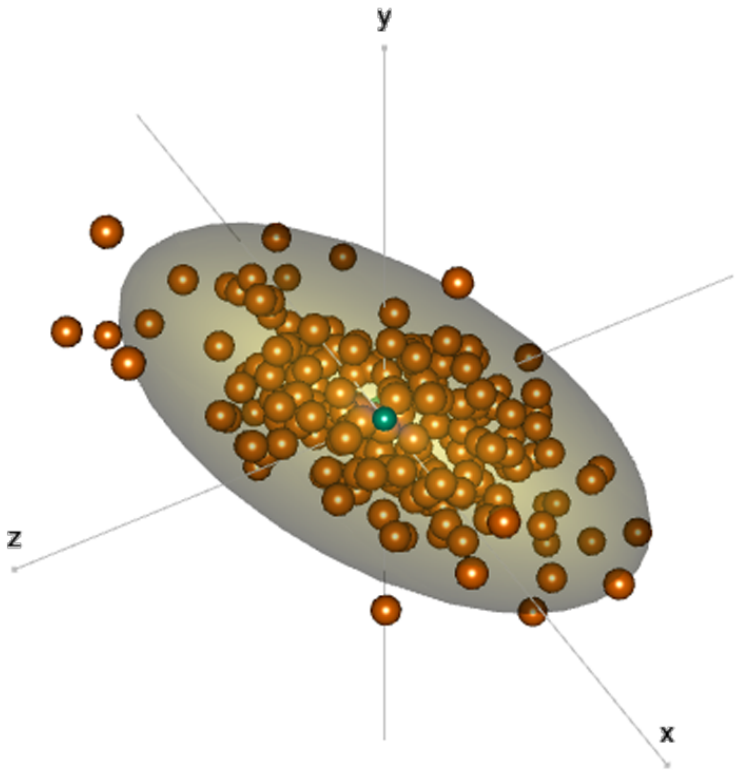


Рис. 6: Проеция элипсоида на плоскость

дящая через фиксированную точку, а в пространстве такие прямые образуют целую плоскость. Чтобы понять, как провести в этой плоскости вторую и третью главные компоненты, спроектируем все наши точки на эту плоскость.

Для этого посмотрим на нашу картинку в направлении только что нарисованной стрелочки (посмотрим на эту стрелочку «в торец»).

Так мы и получим правильную проекцию (рис. 6).

В этой проекции получается картинка, где видно, что большинство точек лежит в вытянутом эллипсе (являющемся проекцией нашего эллипсоида), наклонённом к горизонтали. У этого эллипса есть две оси: большая и маленкая, причём они перпендикулярны. Вторую главную компоненту нужно выбрать таким образом, чтобы соответствующая ей ось шла в том направлении, в котором достигается максимальный раз-

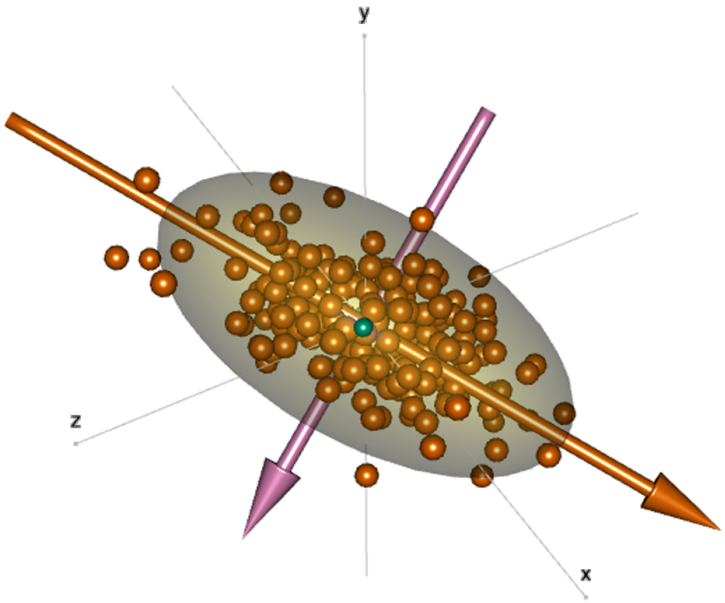


Рис. 7: Оси PC2 PC3

брос точек на нашей проекции. Иными словами, вторую главную компоненту следует направить вдоль длинной оси получающегося эллипса. Третья главная компонента должна быть перпендикулярна второй и значит должна совпадать с малой осью эллипса. Разброс вдоль третьей главной компоненты минимален.

Получается такая картинка (рис.7)

Три получающиеся оси соответствуют трём главным компонентам: зелёная – первой (PC1), оранжевая – второй (PC2) и малиновая – третьей (PC3). Если покрутить картинку, то видно, что точки лежат очень близко к плоскости, проходящей через оси PC1 и PC2: их отклонение от этой плоскости (равное как раз значению PC3) минимальное из всех возможных (Рис.). Это означает, что первые две главные компоненты несут основную часть информации о расположении точек. Если мы «забудем» значения PC3, то есть спроектируем картинку на плоскость, образованную PC1 и PC2, потери в информации будут минимальны.

Тогда если взять только PC1 и PC2, то наша картинка в плоскости, образованной двумя первыми главными компонентами выглядит следу-

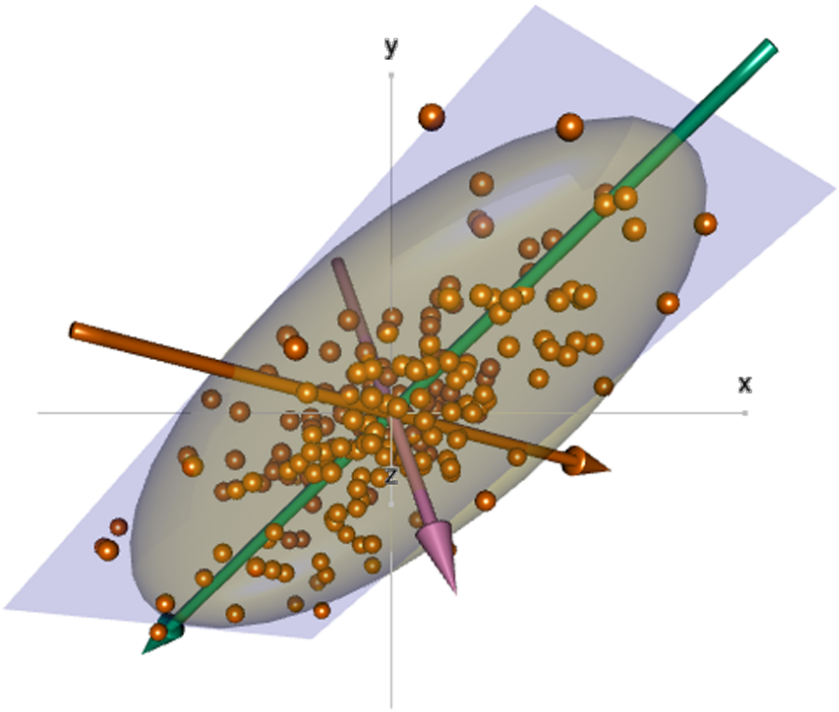


Рис. 8: Оси PC1 PC2 PC3

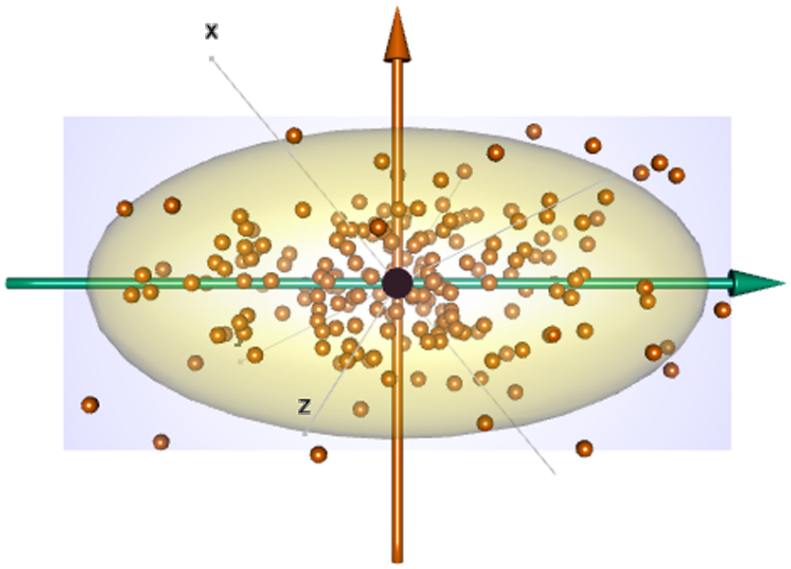


Рис. 9: Данные в плоскости PC1 PC2

ющим образом (рис. 9)

В результате всех этих манипуляций мы сжали трехмерное пространсто в двухмерное и понеряли минимально возможную информацию о исходных данных.

Originalsource

Выше нами расматривался обобщенный математический подход к определению главных компонент. Здесь мы приведем простые математические процедуры получения направлений главных копонент, основанные на нахождении собственных чисел и векторов корреляционной (ковариационной) матрицы.

Для определения собственных чисел и вектров уравнение в матричной записи имеет следующую форму:

$$RV = \lambda V (11)$$

где R -корреляционная матрица, для которой ищется решение; V - искомый собственный вектор, а λ - собственное число.Для матриц неболь-

шого порядка решение базируется на простой форме в виде дискременанта матрицы (в случае больших матриц используются различные итерационные алгоритмы):

$$Det(R - I\lambda) = 0(12)$$

что дает для квадратной матрицы уравнение:

$$Det \begin{pmatrix} 1-\lambda & r_{12} \\ r_{12} & 1-\lambda \end{pmatrix} = 0(13)$$

которое по определению детерминанта может быть представлено в следующем виде:

$$(1-\lambda)(1-\lambda) - r_{12}r_{12} = 0(14)$$

Раскрывая скобки и группируя члены получаем:

$$\lambda^2 - 2\,\lambda + (1 - r_{12}^2)(15)$$

Собственные числа теперь могут быть получены при решении квадратного уравнения. Для двумерной корреляционной матрицы собственные числа имеют вид:

$$\lambda_1 = 1 + r_{12}, (16)$$

$$\lambda_1 = 1 - r_{12}\,.(17)$$

Если между двумя переменными имеется линейная зависимость, то одно собственное число будет 2, а другое -0. Для некоррелированных переменных оба собственных числа равны 1. Заметим также, что сумма собственных чисел $\lambda_1 + \lambda_2 = (1 + r_{12}) + (1 - r_{12}) = 2$ равна числу переменных, а произведение $\lambda_1\lambda_2 = (1 - r_{12}^2$ равна детерминанту корреляционной матрицы. Эти свойства сохраняются для корреляционных матриц любой размерности, причем первое(большее) собственное число представляет величину дисперсии, соотвествующей первой главной компоненте, а второе собственное число-величину дисперсии, соответствующей втрой главной оси и так далее. Так как при использовании корреляционной матрицы сумма собственных чисел равно числу переменных, то разделив первое собственное число на m (число переменных)б можем получить долю дисперсии соотвествующую данному направлению или компоненте:

$$\left(\begin{array}{c}\textbf{Доля соответствующая}\\ \textbf{данной компоненте}\end{array}\right) = \left(\begin{array}{c}\textbf{Соответствующее}\\ \textbf{собственное число}\end{array}\right) \Big/ \mathbf{m} \qquad (18)$$

Рис. 10:

При определении соответствующих собственных векторов есть дополнительное ограничение, состоящее в том, что их длина должна быть единичной. По этой причине коэффициенты нагрузок на главные компонекнты получаются делением коэффициентов собственных векторов на квадратный корень соответствующих собственных чисел, что правильно отражает относительную долю дисперсии наблюдений.

Зная велечину собственных чисел, можно использовать только первые две компоненты, которые обьяснют большую часть дисперсии наблюдений Метод главных компонент не обясняет матрицу корреляций (для это используется факторный анализ), но он применяется для исследования взаимной завичсимости переменных. Заметим, что в случае некоррелированных переменных главных компонент не существует, так как каждой их них соотвествует одинаковая доля дисперсии. Если же корреляция между переменными увеличивается, то доля, объяняемыя несколькими первыми компонентами, возрастает. Это открывает возможности существенного сокращения в описаниии исследуемых объектов и осуществления эффективного визуального анализа многомерных данных.

Реализация PCA с помощью языков программирования R и Python

Продемонстрируем реализацию PCA на Python и R

В качестве исходных данных используем информацию об основных социально-экономических показателях российских регионов за 2018 год. Исходная матрица предикторов имеет размерность 81x16 в качестве выходной переменной используется номинальная переменная тип экономики региона. В реализации на Python переменной "тип экономики"присваевается значение : A-аграрный; D-добывающий, P- промышленный; а при реализации на R добовляется значение: T-торговый.

ОСНОВНЫЕ СОЦИАЛЬНО-ЭКОНОМИЧЕСКИЕ

ПОКАЗАТЕЛИ РЕГИОНОВ РФ 2018 г.

- P1 - Площадь территории,тыс. км2

- P2 - Численность населения на 1 января
2019 г., тыс. человек
- P3 - Среднегодовая численность занятых,
тыс. человек
- P4 - Среднедушевые денежные доходы (в месяц), руб
- P5 - Потребительские расходы в среднем
на душу населения (в месяц), руб
- P6 - Среднемесячная номинальная начисленная
заработная плата работников организаций, руб.
- P7 - Валовой региональный продукт в 2018 г. , млн руб
- P8 - Инвестиции в основной капитал, млн руб
- P9 - Основные фонды в экономике (по полной учетной
стоимости; на конец года),млн руб.
- P10 - Добыча полезных ископаемых ,млн. руб
- P11 - Обрабатывающие производства, млн руб
- P12 - Обеспечение электрической энергией, газом
и паром, млн руб
- P13 - Водоснабжение; водоотведение, организация сбора
и утилизации отходов, млн руб
- P14 - Продукция сельского хозяйства , млн руб
- P15 - Ввод в действие жилых домов, тыс. м2
- P16 - Оборот розничной торговли, млн руб
- P18 - Тип региональной экономики (A,D,P,T)

Код на Python (редукция 3 в 2):

Здесь из исходной таблицы выбираются три столбца P10,P11,P14 , а выходная переменная P18 принимает три значения A,D,P

```python
from mpl_toolkits.mplot3d import Axes3D
import numpy as np
import scipy.stats as st
import matplotlib.pyplot as plt
from pandas import Series, DataFrame
import pandas as pd
from sklearn import decomposition

result =pd.read_table('D:\Machine learning\Py_ML\dan04.txt',sep='\s+')
print(result)
```

```
y=result['P18']

X =DataFrame(result, columns=['P10','P11','P14'])
y=np.array(y)
X=np.array (X)

SUMX=X[:,0] +X[:,1] + X[:,2]
X[:,0] =X[:,0] /SUMX
X[:,1] =X[:,1] /SUMX
X[:,2] =X[:,2] /SUMX

#описательные статистики
print('Описательные статистики предикторов')
for num in range(3) :
  print('Номер переменной: %.0f' %num)
  print('число элентов массива =',len(X[:,num]))
  print('среднее значение =',np.mean(X[:,num]))
  print('стандартное отклонение =',np.std(X[:,num]))
  print('мин мах =',np.min(X[:,num]), np.max(X[:,num]))

 # Преобразование данных , уменьшающее размерность до 2
pca = decomposition.PCA(n_components=2,whiten = True)
pca.fit(X)
X1 = pca.transform(X)

print('Основные оси, представляющие направления максимальной дисперсии')
print(pca.components_)
print(' Величина отклонений, объясняемых PC1,PC2')
print(pca.explained_variance_)
print(' Процент вариации, объясняемый PC1,PC2 ')
print(pca.explained_variance_ratio_)
print('Сингулярные значения PC1,PC2 ')
print(pca.singular_values_)

#построение исходного графика 3 D
fig = plt.figure(1, figsize=(12, 10))
ax = fig.add_subplot(111, projection='3d')
```

```
xs = X[:,0]
ys = X[:,1]
zs =X[:,2]
ax.scatter(xs, ys, zs,c=y, cmap=plt.cm.nipy_spectral, edgecolor='k')
ax.set_title("Исходное пространство предикторов")
ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')

y = np.choose(y, [1, 2, 0]).astype(np.float)
fig = plt.figure(2,figsize=(10, 8))
plt.clf()
plt.cla()
plt.scatter(X1[:, 0], X1[:, 1], c=y, cmap=plt.cm.nipy_spectral, edgecolor='k')
plt.title("Снижение размерности методом PCA")
plt.xlabel("PC1")
plt.ylabel("PC2")
plt.show()
```

Результаты работы кода python представлены ниже

Код на R

Здесь реализуем два варианта. Первый в качестве исходной матрицы предикторов используем значения шестнадцати показателей P1-P16, а во втором четырех P10,P11,P14,P16.В обоих случаях P18 номинальная выходная переменная принимает четыре значения:A,P,T,D

```R
#СНИЖЕНИЕ РАЗМЕРНОСТИ МЕТОД PCA
library(vegan)
library(pvclust)
library(ggplot2)
library(ggthemes)
library(ade4)

rm(list=ls(all=TRUE))

load("D:/Machine learning/ML_scripts/xRegion.RData")
```

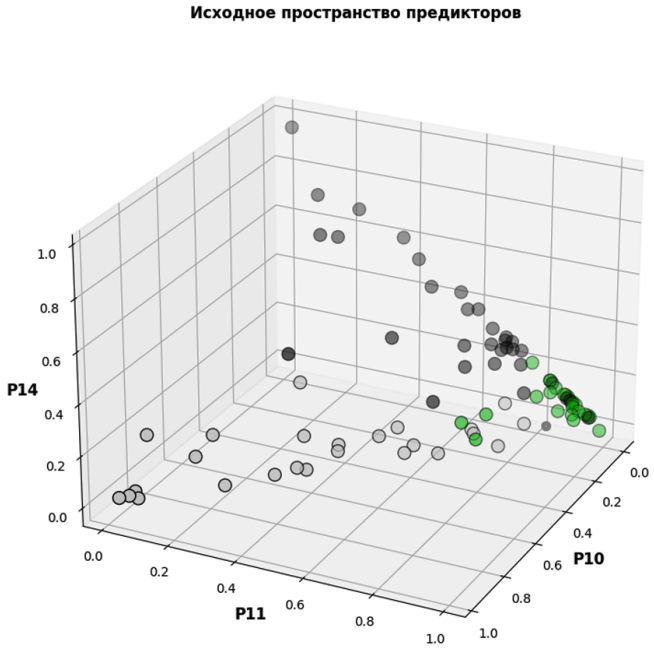


Рис. 11: Данные в исходном пространстве P10,P11,P14

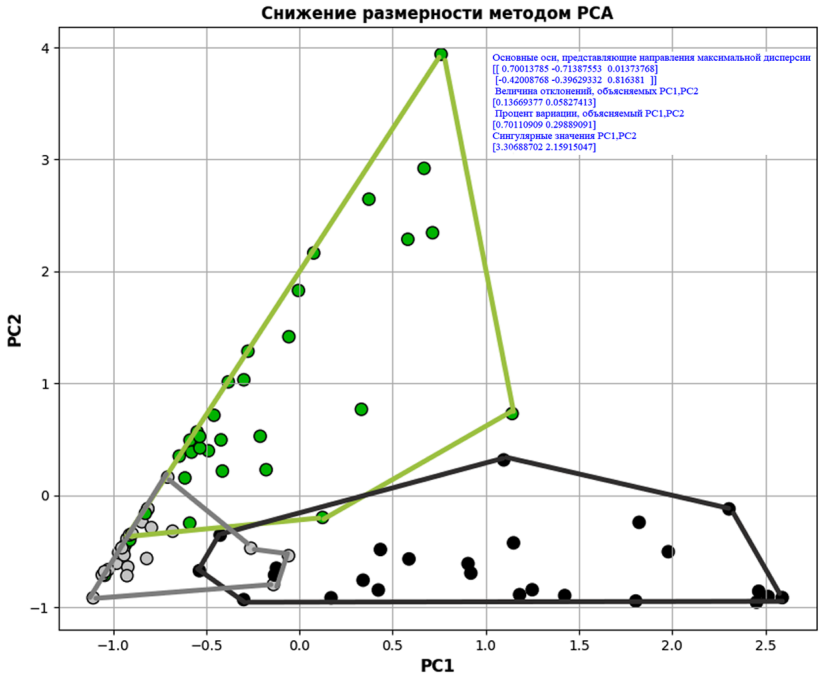


Рис. 12: Данные в плоскости PC1 PC2

```
(x[,1:2])
#ФОРМИРОВАНИЕ data.frame ОСНОВНЫЕ
СОЦИАЛЬНО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ РЕГИОНОВ РФ
xreg <-data.frame(P1=x$Pok_001,P2=x$Pok_002,
P3=x$Pok_003,P4=x$Pok_004,
P5=x$Pok_005,P6=x$Pok_006,P7=x$Pok_007,
P8=x$Pok_008,P9=x$Pok_009,P10=x$Pok_010,
P11=x$Pok_011,P12=x$Pok_012,P13=x$Pok_013,
P14=x$Pok_014,P15=x$Pok_015,P16=x$Pok_016,
P17=x$Pok_017,P18=x$Pok_018, ind =x$IndexReg)
#xreg
#Формирование фактора тип экономики региона
xreg_f <- as.factor(x$Pok_018)
ind_f <-as.factor(x$IndexReg)

#нормирующая функция
norm <-function(x){
return ((x-min(x))/(max(x)-min(x)))
}
xstr_nor <-as.data.frame(lapply(xreg[1:16],norm))
Y <-xstr_nor
xstr_nor <-cbind(xstr_nor,xreg_f,ind_f)
summary(xstr_nor)

#СНИЖЕНИЕ РАЗМЕРНОСТИ ==========

#Метод главных компонент PCA=========
#расчет главных компонент
#эту команду центрирования и нормирования целесообразно
#использовать для первичных не нормированных данных
#xreg_nor.pca <- princomp(scale(xstr_nor[,-17]))

#эту команду целесообразно использовать для
#предворительно нормированных данных
ww=c(-17,-18)
xstr_nor.pca <- princomp(xstr_nor[,ww])

#график главных компонет
```

```
#x11()
plot(xstr_nor.pca)

#основные характеристики компонент
#Standard deviation - стандартное отклонение
#Proportion of Variance - доля вариации
#Cumulative Proportion -суммарная доля
summary(xstr_nor.pca)

#x11()
xstr_nor.p <- predict(xstr_nor.pca)
plot(xstr_nor.p[,1:2], type="n", xlab="PC1", ylab="PC2")
text(xstr_nor.p[,1:2],labels=abbreviate(xstr_nor[,18],1,
 method="both.sides"))

#x11()
biplot(xstr_nor.pca)

loadings(xstr_nor.pca)

#x11()
xstr_nor.dudi <- dudi.pca(xstr_nor[,1:16], scannf=FALSE)
s.class(xstr_nor.dudi$li, xstr_nor[,17])

#ГЛАВНЫЕ КОМПОНЕНТЫ  С
#ИСПОЛЬЗОВАНИЕМ ПАКЕТА vegan
#Формирование компоент и их параметры
mod.pca <- rda(Y ~ 1)
#summary(mod.pca)

#ПРЕДСТАВЛЕНИЕ РЕГИОНОВ В
#ПРОСТРАНСТВЕ ДВУХ ПЕРВЫХ КОМПОНЕНТ
pca.scores <- as.data.frame(summary(mod.pca)$sites[,1:2])
pca.scores <- cbind(pca.scores, xreg_f)

#График регионов по классам в пространстве двух главных компонент 1
#x11()
g3 <-ggplot(data= pca.scores, aes(x=PC1,y=PC2,colour = xreg_f),
```

```
colors = "Accent")
g3 <-g3 +geom_point(size=4)
g3 <-g3 +labs(title ="Регионы в пространсте двух главных компонент",
x="Компонента 1", y="Компонента 2",colour = "Классы")
#g3 <-g3 + theme_fivethirtyeight() +  scale_color_fivethirtyeight()
#g3 <-g3 + theme_stata() + scale_colour_stata()
print(g3)
#График классов регионов в пространстве двух главных компонент 2
# Составляем таблицу для "каркаса" точек на графике
l <- lapply(unique(pca.scores$xreg_f), function(c)
{ f <- subset(pca.scores, xreg_f == c); f[chull(f), ]})
hull <- do.call(rbind, l)
# Включаем в названия осей доли объясненной дисперсии
axX <- paste("PC1 (",
as.integer(100*mod.pca$CA$eig[1]/sum(mod.pca$CA$eig)),"%)")
axY <- paste("PC2 (",
as.integer(100*mod.pca$CA$eig[2]/sum(mod.pca$CA$eig)),"%)")
#x11()
ggplot() +
geom_polygon(data=hull,aes(x=PC1,y=PC2, fill=xreg_f),
alpha=0.4, linetype=0) +
labs(title ="Регионы в пространсте двух главных компонент")+
geom_point(data=pca.scores,aes(x=PC1,y=PC2,shape= xreg_f,
colour=xreg_f),size=3) +
scale_colour_manual( values = c('purple', 'blue','red','green'))+
xlab(axX) + ylab(axY) + coord_equal() + theme_bw()
```

Как видго из графиков во втором варианте анализа за счет выбора значительно меньшего числа информативных признаков и их разумной стандартизации метод PCA позволяет визуально решить задачу кластеризации россйких регионов по типу экономики. При этом реализация на R показала,что классы A и T сильно пересекаются и их целесообразно объединить в один класс с условным название A. Реализация данного варианта в коде на python показала, что пересечений кластеров становиться меньше. В цеолом метод PCA является проверенным, обоснованным и эффективным методом снижения размерномти матрицы предикторов для эмпирических моделей.

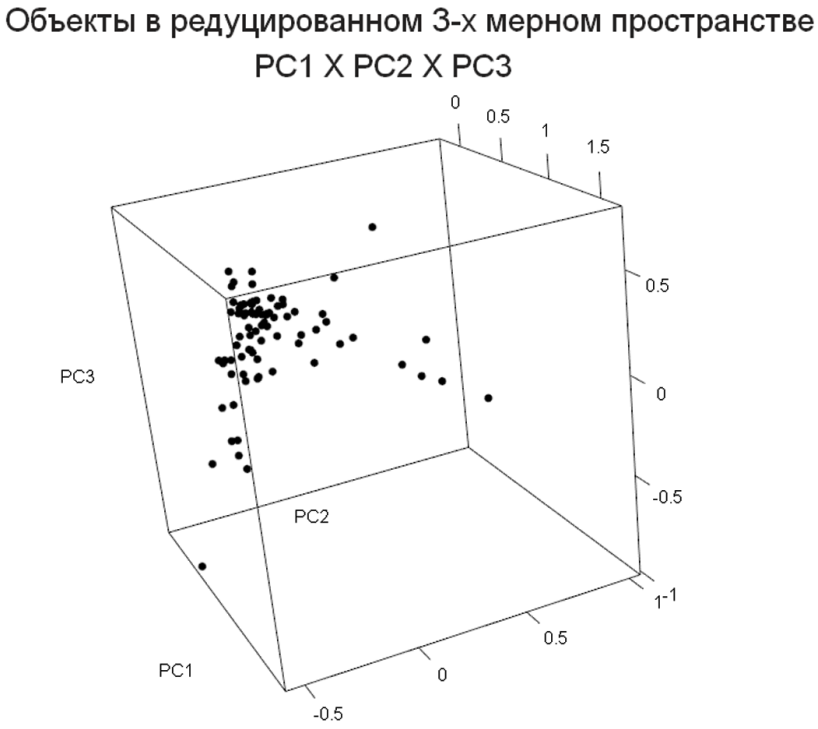
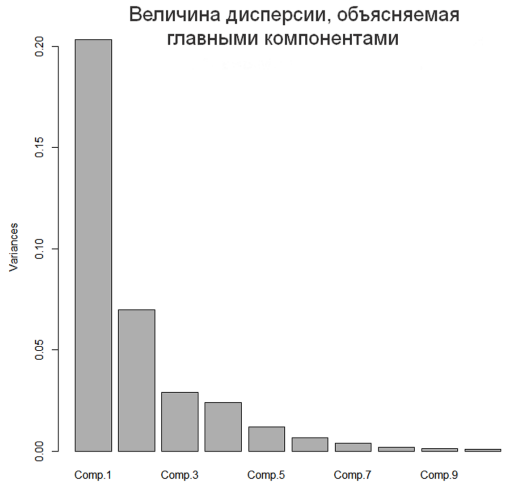
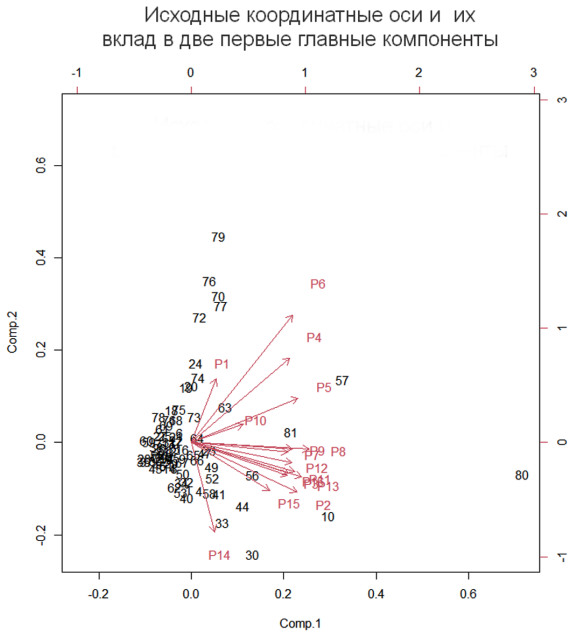


Рис. 13: Данные в PC1 PC2 PC3




```
Importance of components:
                          Comp.1     Comp.2
Standard deviation     0.4509130 0.2642522
Proportion of Variance 0.5752066 0.1975494
Cumulative Proportion  0.5752066 0.7727560
                          Comp.3     Comp.4
Standard deviation     0.17003252 0.1545530
Proportion of Variance 0.08179039 0.0675761
Cumulative Proportion  0.85454642 0.9221225
                          Comp.5     Comp.6
Standard deviation     0.10850118 0.08153977
Proportion of Variance 0.03330483 0.01880950
Cumulative Proportion  0.95542735 0.97423685
                          Comp.7      Comp.8
Standard deviation     0.06247581 0.041204292
Proportion of Variance 0.01104237 0.004803118
Cumulative Proportion  0.98527921 0.990082332
                          Comp.9      Comp.10
Standard deviation     0.033956117 0.031141201
Proportion of Variance 0.003261928 0.002743526
Cumulative Proportion  0.993344260 0.996087785
                          Comp.11      Comp.12
Standard deviation     0.025122839 0.019400524
Proportion of Variance 0.001785565 0.001064793
Cumulative Proportion  0.997873351 0.998938144
                          Comp.13      Comp.14
Standard deviation     0.0142753389 0.0092184657
Proportion of Variance 0.0005765157 0.0002404117
Cumulative Proportion  0.9995146596 0.9997550713
                          Comp.15      Comp.16
Standard deviation     0.0086367503 3.461692e-03
Proportion of Variance 0.0002110275 3.390121e-05
Cumulative Proportion  0.9999660988 1.000000e+00
```

Рис. 14: Важность главных компонент (вариант 16)




```
Loadings:
    Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6
P1          0.293         0.665  0.641  0.178
P2   0.285 -0.229                0.174
P3   0.259 -0.152                       0.205
P4   0.265  0.389  0.319 -0.169 -0.138
P5   0.288  0.205  0.295 -0.140         0.186
P6   0.273  0.587  0.185              -0.134
P7   0.260        -0.198        -0.109  0.153
P8   0.317        -0.223  0.260 -0.232 -0.190
P9   0.273        -0.239  0.111 -0.231
P10  0.139        -0.227  0.449 -0.416 -0.298
P11  0.278 -0.135        -0.112         0.111
P12  0.272        -0.138                0.152
P13  0.297 -0.160        -0.123  0.207
P14         -0.411  0.695  0.408 -0.270  0.194
P15  0.211 -0.221  0.241         0.303 -0.793
P16  0.264 -0.141        -0.127  0.108  0.125
    Comp.7 Comp.8 Comp.9 Comp.10 Comp.11 Comp.12
P1           0.112
P2           0.187 -0.203 -0.437   0.493  -0.257
P3  -0.102  0.208        -0.103   0.269
P4                0.733 -0.240   0.102
P5   0.804        -0.276
P6  -0.483        -0.506  0.148
P7           0.207         0.170           0.312
P8           0.174  0.125  0.303          -0.233
P9           0.201  0.103         -0.215   0.190
P10  0.106 -0.277 -0.124 -0.381   0.184
P11         -0.540         0.555   0.414
P12         -0.125                -0.526  -0.699
P13         -0.529        -0.334  -0.318   0.388
P14 -0.203
P15  0.113  0.137         0.128
P16         0.344                 -0.132   0.290
    Comp.13 Comp.14 Comp.15 Comp.16
P1
P2           0.147   0.233  -0.409
P3                  -0.583   0.598
P4
P5
P6
P7   0.257  -0.197  -0.458  -0.598
P8  -0.659  -0.224
P9           0.770   0.227
P10  0.337  -0.191          0.149
P11  0.227           0.159
P12  0.255
P13 -0.411
P14
P15  0.193          -0.141
P16  0.211  -0.489   0.530   0.274
```

Рис. 15: Вклад показателй в главные компоненты (вариант 16)

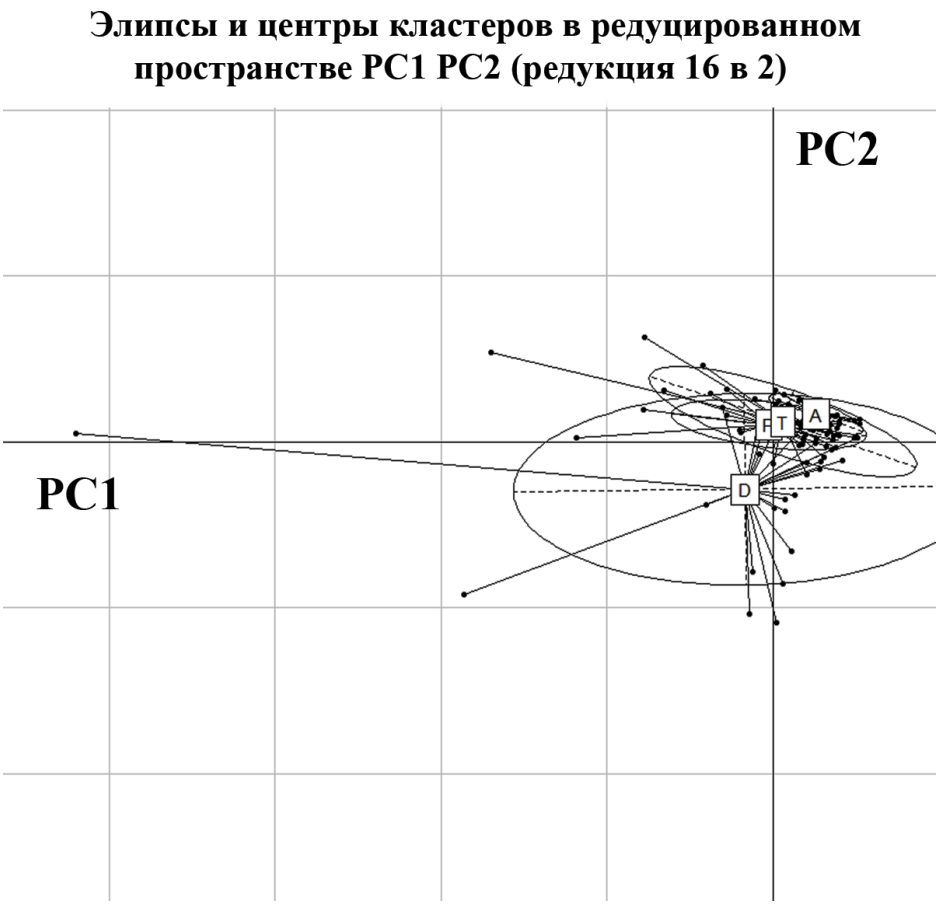


Рис. 16: Кластеры в пространстве PC1,PC2 (вариант 16)

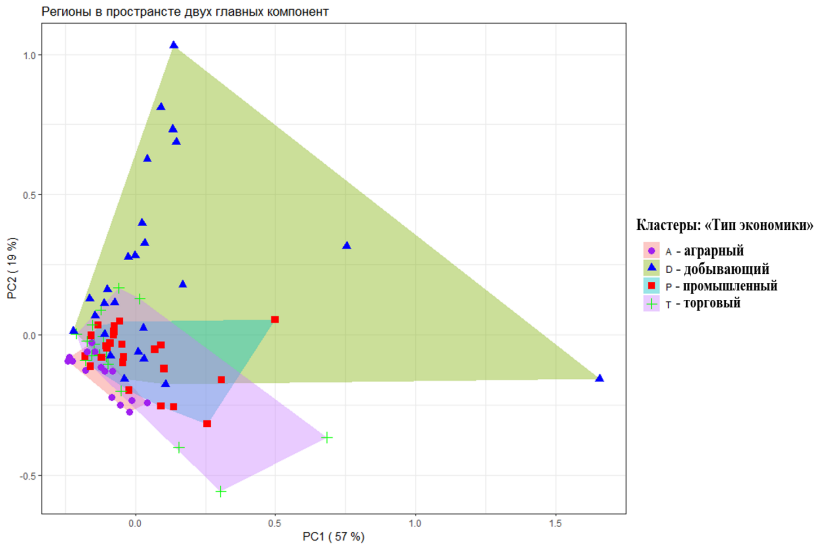
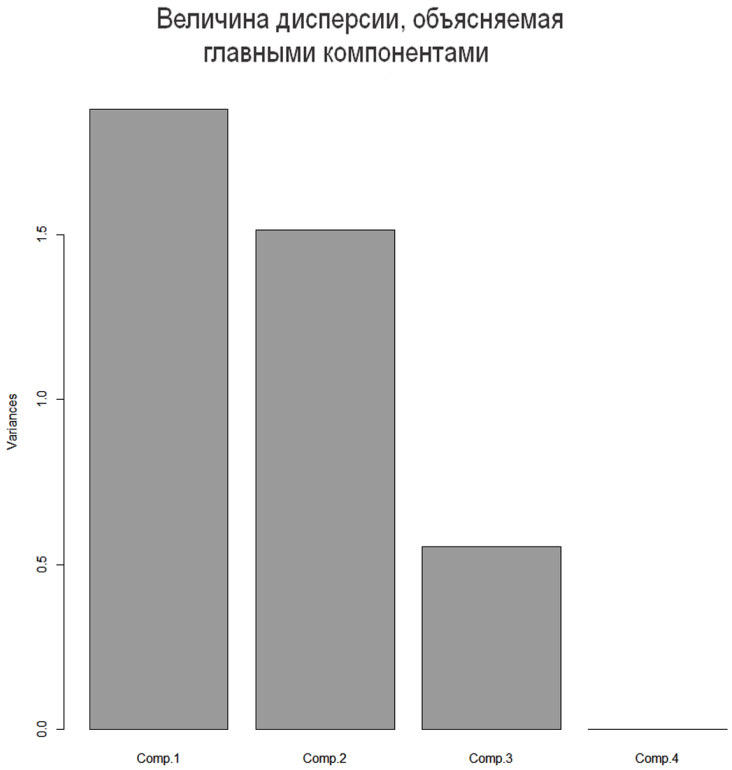


Рис. 17: Плоские выпуклые многогоугольники, натянутые на кластеры (вариант 16)




```
Importance of components:
                          Comp.1    Comp.2    Comp.3       Comp.4
Standard deviation     1.3714116 1.2311170 0.7444450 2.980232e-08
Proportion of Variance 0.4760698 0.3836487 0.1402815 2.248202e-16
Cumulative Proportion  0.4760698 0.8597185 1.0000000 1.000000e+00
```

Рис. 18: [Важность главных компонент (вариант 4)

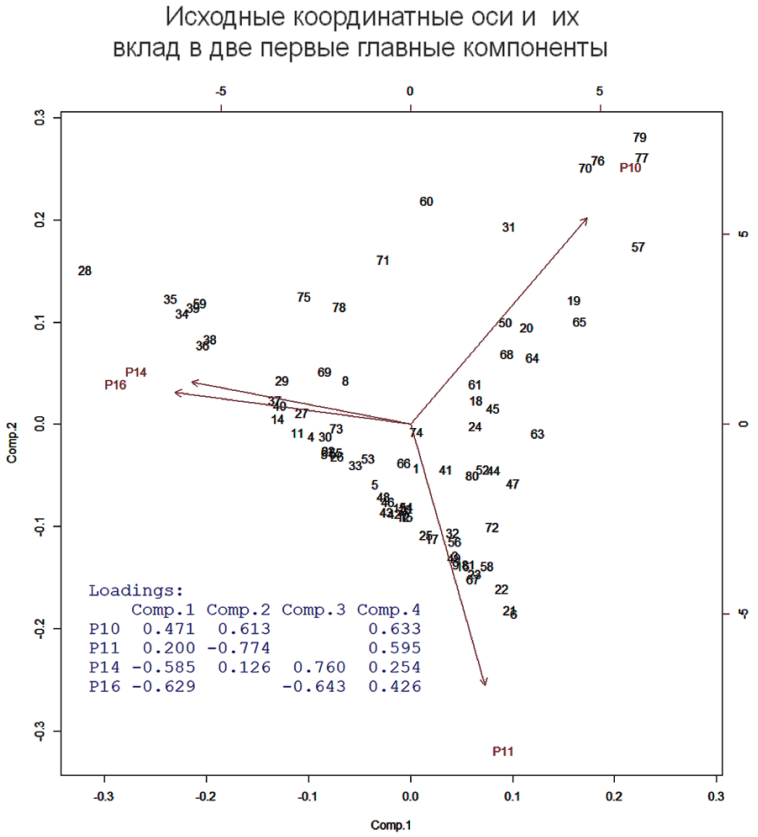


Рис. 19: Вклад показателй в главные компоненты (вариант 4)

Рис. 20: Кластеры в пространстве PC1,PC2 (вариант 4)

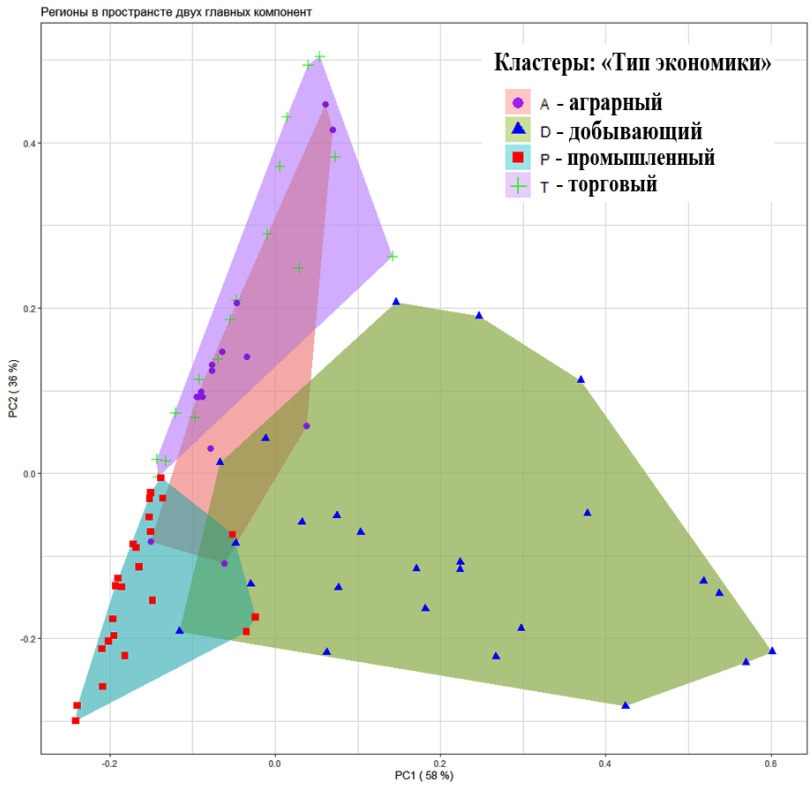


Рис. 21: Плоские выпуклые многогоугольники, натянутые на кластеры (вариант 4)