Wspomaganie decyzji w warunkach ryzyka
WDWR
projekt 20204

Aleksandra Horubała

Problem

Rozważamy następujące zagadnienie planowania produkcji:

- Realizacja umowy wymaga dostawy 1100 sztuk komponentu A oraz 1200 sztuk komponentu B po upływie okresu 3 miesięcy.
- Koszty produkcji komponentów (zł/szt.) określają składowe wektora losowego $R = (R_1, \dots, R_6)^T$:

	Miesiąc 1 (M1)	Miesiąc 2 (M2)	Miesiąc 3 (M3)
A	R_1	R_2	R_3
B	R_4	R_5	R_6

- Wektor losowy R opisuje 6-wymiarowy rozkład t-Studenta z 5 stopniami swobody, którego wartości składowych zostały zawężone do przedziału [20; 60]. Parametry μ oraz Σ niezawężonego rozkładu t-Studenta są następujące:

$$\mu = \begin{bmatrix} 55 \\ 40 \\ 50 \\ 35 \\ 45 \\ 30 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} 1 & 1 & 0 & 2 & -1 & -1 \\ 1 & 16 & -6 & -6 & -2 & 12 \\ 0 & -6 & 4 & 2 & -2 & -5 \\ 2 & -6 & 2 & 25 & 0 & -17 \\ -1 & -2 & -2 & 0 & 9 & -5 \\ -1 & 12 & -5 & -17 & -5 & 36 \end{bmatrix}.$$

- Firma może składować do 150 sztuk komponentu A z miesiąca na miesiąc nie ponosząc żadnych kosztów. Po przekroczeniu tej liczby koszt składowania komponentów A kształtuje się na poziomie 15% miesięcznych kosztów wytwarzania. Analogicznie dla komponentu B.
- Firma w celu wytworzenia komponentów potrzebuje zasobów pozyskiwanych z zewnątrz. Szczegóły dostępnych dostaw i wymagań są następujące:

Zasób	Potrzebne na sztukę A	Potrzebne na sztukę B	Możliwe dostawy M1	Możliwe dostawy M2	Możliwe dostawy M3
Z1	0,2	0,7	600	700	550
Z2	0,8	0,3	1400	900	1200

1 Zadanie 1

Treść zadania: Zaproponować jednokryterialny model wyboru w warunkach ryzyka z wartością oczekiwaną jako miarą kosztu. Wyznaczyć rozwiązanie optymalne.

1.1 Interpretacja treści zadania

Cel zadania: Celem zadania jest zaproponowanie jednokryterialnego modelu minimalizującego wartość oczekiwaną funkcji kosztu f:

$$min\{\mathbb{E}(f(\mathbf{X})) \mid \mathbf{X} \in Q\}.$$

Z treści zadania rozumiem, że produkty są dostarczane klientowi po upływie 3 miesięcy. Do tego czasu muszą być składowane w fabryce. Jeżeli na koniec miesiąca liczba wytworzonych produktów jest większa niż wartość maksymalna (150) naliczana jest dodatkowa opłata wynosząca 15% kosztów wytwarzania w danym miesiącu. Rozumiem, że gdyby wytworzono wszystkie produkty w pierwszym miesiącu, trzeba by je składować przez kolejne dwa miesiące i płacić 15% kosztów wytwarzania we wszystkich trzech miesiącach.

1.2 Obliczenie wartości oczekiwanych wielowymiarowego rozkładu studenta

Wartości oczekiwana 6-wymiarowego rozkładu t-Studenta, zawężonego do przedziału [20;60] obliczono zgodnie ze wzorem podanym przez prowadzącego (zachowując oznaczania z materiałów do wykładu)

$$\mathbb{E}(R) = \mu + \sigma \cdot \frac{\Gamma((v-1)/2)(v+a^2)^{-(v-1)/2} - (v+b^2)^{-(v-1)/2} v^{v/2}}{2(F_v(b) - F_v(a))\Gamma(v/2)\Gamma(1/2)}.$$

Dla parametrów z zadania wartości te wynoszą:

$$\begin{aligned}
\mathbb{E}(R_1) &= 54.9735999193,\\
\mathbb{E}(R_2) &= 40.0,\\
\mathbb{E}(R_3) &= 49.9480666349,\\
\mathbb{E}(R_4) &= 35.4816698893,\\
\mathbb{E}(R_5) &= 44.9336086285,\\
\mathbb{E}(R_6) &= 32.3866043203.
\end{aligned}$$

Zakładamy, że tyle wynoszą koszty produkcji w tym punkcie zadania. Zatem koszty produkcji przedstawia tabela:

	Miesiąc 1 (M1)	Miesiąc 2 (M2)	Miesiąc 3 (M3)
A	54.9735999193	40.0	49.9480666349
B	35.4816698893	44.9336086285	32.3866043203

Ze względu na swoją liniowość, wartość oczekiwana całkowitego kosztu produkcji jest sumą iloczynów liczby wytworzonych produktów i wartości oczekiwanych kosztów produkcji i przechowywania w poszczególnych miesiącach. Problem decyzyjny polega na dokonaniu wyboru w którym miesiącu tworzyć jaką liczbę produktów. Problem opisano w języku modelowania AMPL, po czym rozwiązano przy pomocy solvera CPLEX i otrzymano optymalną decyzję.

1.3 Optymalne rozwiązanie problemu

Aby obliczyć optymalne rozwiązanie problemu zaprogramowano kod w języku Python wykorzystujący API do języka AMPL i języka R. Kod programu umieszczono na *githubie* pod adresem https://github.com/keIpie/wdwr. Zdefiniowano model w języku AMPL opisany w Sekcji 2.3, który został wczytany przez program. Zbiory i parametry modelu są dynamicznie wypełniane w programie. Na ich podstawie generowane są układy równań, do rozwiązania których wykorzystano solver *CPLEX*.

Znalezione optymalne rozwiązanie ma minimalny koszt równy 105263.37439251498 zł. Optymalne liczby produktów, które opłacało się wytworzyć prezentuje tabela

	Miesiąc 1 (M1)	Miesiąc 2 (M2)	Miesiąc 3 (M3)
A	0	1100	0
B	415	0	785

Wybór jest zgodny z intuicją: jeśli chodzi o produkt A koszty wytwarzania w drugim miesiącu są zdecydowanie najniższe, różnica kosztów jest większa niż dodatkowy koszt przechowywania w tym miesiącu. W przypadku produktu B koszt wytwarzania w drugim miesiącu jest zdecydowanie największy, warto więc jak najwięcej produktów wytworzyć w miesiącu ostatnim (wtedy koszt przechowywania jest najmniejszy). Ponadto ograniczenia na zasoby pozwalają zrealizować takie decyzje.

1.4 Opis modelu

Model problemu napisano w języku modelowanie AMPL. Poszczególne fragmenty modelu opisano w poniższych podrozdziałach.

1.4.1 Definicja zbiorów

Zdefiniowano trzy zbiory:

- PRODUCTS (P) - zbiór wytwarzanych produktów (produkty A i B),
- MONTHS (M) - zbiór miesięcy (miesiące M1, M2 i M3),
- RESOURCES (R) - zbiór zasobów (zawierający zasoby Z1 i Z2).

1.4.2 Definicja parametrów

Zdefiniowano parametry problemu:

- promised - lista zawierająca liczbę zamówień na produkty A i B ([1100,1200]),
- maxno - maksymalna liczba produktów, która może być przechowywana nie ponosząc dodatkowych kosztów (150),
- percent - procent określający dodatkowy koszt przechowywania produktów, których wybradukowano więcej niż *maxno* (15),
- cost - dwuwymiarowa tablica kosztów, przechowująca koszt wytwarzania danego produktu w danym miesiącu (obliczone wartości oczekiwane),
- demand - dwuwymiarowa tablica, przechowująca zapotrzebowania na zasoby dla poszczeghólnych produktów,
- delivery - dwuwymiarowa tablica, przechowująca możliwe liczności dostaw zasobów w konkretnych miesiącach,
- chronology - dwuwymiarowa tablica, przechowująca wartości binarne, indeksowana miesiącami na obu współrzędnych; opisuje który miesiąc był wcześniej;

1.4.3 Definicja zmiennych

Zdefiniowano następujące zmienne:

- Made - dwuwymiarowa tablica nieujemnych zmiennych całkowitych, przechowująca liczby wytworzonych produktów do danego miesiąca (wartości obliczane przy pomocy danych o chronologii miesięcy),
- Make - dwuwymiarowa tablica nieujemnych zmiennych całkowitych, przechowująca liczby wytworzonych produktów w danym miesiącu,
- cond - dwuwymiarowa tablica zmiennych binarnych, przechowująca wartość 0 jeżeli do danego miesiąca wyprodukowano mniej niż *maxno* danego produktu, a 1 w przeciwnym wypadku,
- extra_stored - nieujemna zmienna całkowita opisująca liczbę produktów, za których przechowywanie trzeba dodatkowo płacić; została wprowadzona, żeby usunąć nieliniowość, którą wprowadza zmienna warunkowa *cond*;

1.4.4 Definicja funkcji celu

Optymalizacja polega na minimalizacji funkcji celu, którą stanowi wartość oczekiwana funkcji kosztu. W tym problemie funkcja kosztu składa się z dwóch części: kosztów produkcji i kosztów przechowywania

$$\sum_{p\in P}\sum_{m\in M} koszt\_produkcji[p,m] + koszt\_przechowywania[p,m] =$$

$$\sum_{p\in P}\sum_{m\in M} cost[p,m] * Make[p,m] + extra\_stored * percent * cost[p,m].$$

Pierwszy człon funkcji stanowi koszt przechowywania liczony jako $cost[p,m] * Make[p,m]$. Jest to liczba wytworzonych produktów w ramach danego miesiąca przemnożona przez koszt wytwarzania w danym miesiącu. Drugi człon stanowi koszt przechowywania liczony jako $extra\_stored * percent * cost[p,m]$. Oznacza liczbę produktów, za przechowywanie których trzeba dodatkowo płacić wymnożoną przez koszt przechowywania, który w każdym miesiącu wnosi tyle ile koszt wytwarzania przeskalowany przez procent opisany zmienną *percent*.

1.4.5 Definicja ograniczeń

Zdefiniowano następujące ograniczenia:

- Sumaryczna liczba wytworzonych produktów musi być większa od obiecanej klientowi
$$\sum_{m\in M} Make[p,m] >= promised[p].$$
- W każdym miesiącu liczba wytworzonych produktów przemnożona przez stopień wykorzystania zasobu na sztukę produktu nie może przekroczyć liczby dostępnych zasobów w zadanym miesiącu:
$$\sum_{p\in P} Make[p,m] * demand[p,r] <= delivery[m,r],$$
- Wytworzone do tej pory produkty obliczane są dzięki zmiennej opisującej chronologię kolejnych miesięcy: wartość wytworzonych produktów *Make* jest dodawana do zmiennej *Made* dla miesięcy wcześniejszych lub równych danemu miesiącowi
$$Made[p,m] = \sum_{n\in M} Make[p,n] * chronology[m,n];$$
- Liczba wytworzonych do tej pory produktów *Made* musi być większa niż wartość *maxno* przemnożona przez zmienną warunkową *cond* opisującą, czy do tej pory wytworzona maksymalną liczbę produktów:
$$Made[p,m] >= maxno * cond[p,m].$$
Jeżeli wytworzono więcej niż *maxno* produktów zmienna *cond* wynosi 1 i dostajemy prawdziwą nierówność
$$Made[p,m] >= maxno.$$

Jeżeli do tej pory nie wytworzono *maxno* produktów zmienna *cond* wynosi 0 i dostajemy prawdziwą nierówność

$$Made[p, m] >= 0.$$

- Liczba wytworzonych do tej pory produktów *Made* musi być mniejsza lub równa wartości

$$Made[p, m] <= maxno + cond[p, m] * (promised[p] - maxno).$$

Jeżeli do tej pory wytworzono więcej niż *maxno* produktów zmienna *cond* wynosi 1 i dostajemy prawdziwą nierówność

$$Made[p, m] <= promised[p].$$

Jeżeli do tej pory nie wytworzono *maxno* produktów zmienna *cond* wynosi 0 i dostajemy prawdziwą nierówność

$$Made[p, m] <= maxno.$$

Na tym etapie pojawia się problem z obliczeniem liczby produktów, za których przechowywanie trzeba dodatkowo płacić. Liczbę tę można by opisać następująco

$$extra\_stored[p, m] = cond[p, m] * (Made[p, m] - maxno)$$

jednak takie ograniczenie nie jest liniowe. Dlatego zastąpiono je ograniczeniami:

- zmienna *extra_stored* musi być większa lub równa zero, nie może też przekroczyć maksymalnej liczby przechowywanych produktów, pomniejszonej o liczbę produktów, które można przechowywać za darmo:

$$0 <= extra\_stored[p, m] <= promised[p] - maxno$$

- kolejne ograniczenie jest postaci

$$(Made[p, m]-maxno)-extra\_stored[p, m]+cond[p, m]*(promised[p]-maxno) <=$$

$$<= promised[p] - maxno.$$

Zauważmy, że jeżeli $cond[p, m] = 1$ (mamy więcej niż *maxno* produktów) to ograniczenie ma postać

$$(Made[p, m] - maxno) <= extra\_stored[p, m],$$

a więc liczba dodatkowo przechowywanych produktów jest mniejsza lub wartości *extra_stored*. Zauważmy, że ze względu na postać minimalizowanej funkcji zawsze wybierana będzie minimalna możliwa wartość zmiennej *extra_stored*, która wynosi dokładnie $Made[p, m] - maxno$, a więc jest równa liczbie nadmiarowych produktów.

Jeżeli $cond[p,m] = 0$ (mamy mniej niż *maxno* produktów) to ograniczenie ma postać

$$Made[p,m] - extra\_stored[p,m] <= promised[p]$$

co jest prawdą dla każdej nieujemnej wartości *extra_stored*, ponieważ zawsze zachodzi $Made[p,m] <= promised[p]$. Zatem ze względu na postać minimalizowanej funkcji zmienna *extra_stored* zawsze otrzyma najmniejszą możliwą wartość czyli 0.

2 Zadanie 2

Treść zadania: Jako rozszerzenie powyższego zaproponować dwukryterialny model kosztu i ryzyka z wartością oczekiwaną jako miarą kosztu i odchyleniem przeciętnym jako miarą ryzyka. Dla decyzji $x \in Q$ odchylenie przeciętne jest definiowane jako $\delta(x) = \sum_{t=1}^{T} |\mu(x) - r_t(x)| \cdot p_t$, gdzie $\mu(x)$ oznacza wartość oczekiwaną, $r_t(x)$ realizację dla scenariusza t, p_t prawdopodobieństwo scenariusza t.

a. Wyznaczyć obraz zbioru rozwiązań efektywnych w przestrzeni ryzyko–koszt.

b. Wskazać rozwiązania efektywne minimalnego ryzyka i minimalnego kosztu. Jakie odpowiadają im wartości w przestrzeni ryzyko–koszt?

c. Wybrać trzy dowolne rozwiązania efektywne. Sprawdzić czy zachodzi pomiędzy nimi relacja dominacji stochastycznej pierwszego rzędu. Wyniki skomentować, odnieść do ogólnego przypadku.

2.1 Interpretacja treści zadania

Cel zadania: Celem zadania jest zaproponowanie dwukryterialnego modelu minimalizującego wartość oczekiwaną funkcji kosztu f oraz ryzyko δ liczone jako odchylenie przeciętne:

$$min\{(\mathbb{E}(f(\mathbf{X})), \delta(x)) \mid \mathbf{X} \in Q\}.$$

Aby rozwiązać zadanie optymalizacji przy pomocy programowania liniowego zdecydowano się sprowadzić model dwukryterialny do jednokryterialnego przy pomocy funkcji skalaryzacji.

Z wykładu wiemy, że jeżeli funkcja skalaryzująca s jest ściśle rosnąca po współrzędnych, to rozwiązanie optymalne skalaryzacji jest rozwiązaniem efektywnym zadania wielokryterialnego. Jako funkcję skalaryzacji zdecydowano się przyjąć funkcję

$$s(f_1(x), f_2(x)) = f_1(x) + \lambda \cdot f_2(x),$$

gdzie $\lambda > 0$ jest miarą awersji do ryzyka. Zauważmy, że przy zadaniu minimalizacji chcemy otrzymać jak najmniejszą wartość $s(f_1(x), f_2(x))$. Zatem im większe ustalimy λ (większa chęć uniknięcia ryzyka) tym większa będzie wartość funkcji celu . Chcąc unikać ryzyka, decydujemy się na potencjalnie większe koszty. Dla funkcji z zadania skalaryzacja ma postać

$$s(\mathbb{E}(f(\mathbf{X})), \delta(x)) = \mathbb{E}(f(\mathbf{X})) + \lambda \cdot \delta(x).$$

Po pierwsze zauważmy, że wybrana skalaryzacja jest funkcją rosnącą po współrzędnych, zatem otrzymane rozwiązania będą rozwiązaniami efektywnymi.

Zwróćmy uwagę, że funkcja opisująca odchylenie przeciętne

$$\delta(x) = \sum_{t=1}^{T} |\mu(x) - r_t(x)| \cdot p_t$$

nie jest funkcją liniową. Aby rozwiązać model należy ją zlinearyzować.

Ponieważ wielowymiarowy rozkład t-Studenta jest rozkładem ciągłym, prawdopodobieństwo dowolnej konkretnej wartości kosztów jest równe 0 (ponieważ prawdopodobieństwo zajścia konkretnego zdarzenia opisane rozkładem ciągłym musi wynosić 0). Za możliwe scenariusze zdecydowano się zatem przyjąć punkty losowane z rozkładu t-Studenta. Ze względów obliczeniowych zdecydowano się rozważać $T = 2^{12}$ scenariuszy. Przyjęto, że wszystkie scenariusze są równoprawdopodobne, a ich prawdopodobieństwo wynosi $p_t = \frac{1}{2^{12}}$.

2.2 Optymalne rozwiązanie problemu

Zdefiniowano model liniowy minimalizujący wartość skalaryzacji zadanego modelu dwukryterialnego. Model rozwiązywany jest dla różnych parametrów awersji do ryzyka λ.

2.3 Opis modelu

Model problemu napisano w języku modelowanie AMPL. Poszczególne fragmenty modelu opisano w poniższych podrozdziałach.

2.3.1 Definicja zbiorów

Zdefiniowano trzy zbiory:

- PRODUCTS (P) - zbiór wytwarzanych produktów (produkty A i B),
- MONTHS (M) - zbiór miesięcy (miesiące M1, M2 i M3),
- RESOURCES (R) - zbiór zasobów (zawierający zasoby Z1 i Z2),
- SCENARIOS (S) - zbiór scenariuszy (zawierający 2^{12} scenariuszy).

2.3.2 Definicja parametrów

Poza parametrami zdefiniowanymi w modelu jednokryterialnym, użyto nowych parametrów:

- lmbd - liczba wymierna pomiędzy 0 a 10, opisująca awersję do ryzyka,
- costS - koszty produkcji produktów w poszczególnych miesiącach dla poszczególnych scenariuszy,
- prob - prawdopodobieństwo zajścia scenariusza (takie samo dla wszystkich scenariuszy $\frac{1}{2^{12}}$),

- min_diff - ograniczenie dolne na wartość różnicy $\mu(x) - r_t(x)$ (potrzebne do linearyzacji odchylenia przeciętnego),
- max_diff - ograniczenie górne na wartość różnicy $\mu(x) - r_t(x)$ (potrzebne do linearyzacji odchylenia przeciętnego).

2.3.3 Definicja zmiennych

Zdefiniowano następujące dodatkowe zmienne:

- expected_cost - dodatnia zmienna opisująca wartość oczekiwaną kosztu produkcji,
- cost_difference - tablica zmiennych $(\mu(x)-r_t(x)|)$, które mogą być dodatnie lub ujemne,
- cond2 - tablica zmiennych binarnych, które mają wartość 0 jeżeli różnica kosztów dla danego scenariusza (*cost_difference*) jest dodatnia i wartość 1 jeżeli jest ujemna,
- cost_difference_cond2 - tablica niedodatnich zmiennych przechowujących wartość 0 jeżeli różnica kosztów dla danego scenariusza (*cost_difference*) jest dodatnia i wartość różnicy *cost_difference* jeżeli jest ujemna,
- risk - tablica zmiennych przechowująca wartości $|\mu(x) - r_t(x)|$,
- risk_sum - dodatnia zmienna przechowująca łączną wartość ryzyka.

2.3.4 Definicja funkcji celu

Optymalizacja polega na minimalizacji funkcji celu, którą stanowi wartość oczekiwana funkcji kosztu. W tym problemie funkcja kosztu składa się z dwóch części: oczekiwanych kosztów produkcji oraz sumarycznej wartości ryzyka przeskalowanej przez prawdopodobieństwo scenariusza i wartość parametru awersji do ryzyka

$$expected\_cost + lmbd \cdot prob \cdot risk\_sum$$

Pierwszy człon funkcji liczony jest tak samo jak w zadaniu pierwszym. Drugi człon opisuje ryzyko zdefiniowane przez odchylenie przeciętne i został opisany w taki sposób aby model miał postać liniową. Wykonano to definiując wiele dodatkowych ograniczeń.

2.3.5 Definicja ograniczeń

Zdefiniowano następujące nowe ograniczenia:

- Sumaryczna ryzyko jest sumą ryzyk dla poszczególnych scenariuszy

$$risk\_sum = \sum_{s \in S} risk[s],$$

- Oczekiwany koszt jest liczony jak w pierwszym zadaniu

  $$expected\_cost = \sum_{p\in P, m\in M} cost[p,m]\cdot(Make[p,m]+extra\_stored[p,m]\cdot percent),$$

- Różnica *cost_difference* jest liczona:

  $$cost\_difference[s] = \sum_{p\in P, m\in M} (cost[p,m] - costS[p,m,s])\cdot$$

  $$\cdot(Make[p,m] + extra\_stored[p,m]\cdot percent).$$

Na tym etapie pojawia się problem z nieliniowością odchylenia przeciętnego. Chcielibyśmy zadać ryzyko poprzez różnice

$$risk[s] = |cost\_difference[s]| = cost\_difference[s]\cdot(1 - 2\cdot cond2[s])$$

jednak takie ograniczenie nie jest liniowe. Dlatego zastąpiono je ograniczeniami:

- iloczyn $cost\_difference[s]\cdot cond2[s]$ zastąpiono zmienną *cost_difference_cond2[s]*, wtedy ryzyko można opisać w sposób liniowy

  $$risk[s] = cost\_difference[s] - 2\cdot cost\_difference\_cond2[s]$$

- ryzyko dla scenariusza jest ograniczone z dołu przez 0, a z góry przez maksymalne ryzyko (największa możliwa różnica pomiędzy wartością oczekiwaną, a realizacją danego scenariusza):

  $$risk[s] <= max\_diff,$$

- warunkowa różnica kosztów jest większa niż najmniejsza możliwa różnica przemnożona przez zmienną binarną *cond2*

  $$cost\_difference\_cond2[s] >= min\_diff\cdot cond2[s],$$

  faktycznie: jeżeli $cond2[s] = 0$, to zgodnie z definicją $cost\_difference\_cond2[s] >= 0$, a jeżeli $cond2[s] = 1$ to zachodzi trywialna nierówność $cost\_difference\_cond2[s] >= min\_diff\cdot cond2[s]$.

- warunkowa różnica kosztów jest mniejsza niż różnica kosztów (ponieważ dodatnie różnice zostaną zamienione na 0, a ujemne pozostaną takie same)

  $$cost\_difference\_cond2[s] <= cost\_difference[s]$$

- ponadto zachodzi ograniczenie

  $$cost\_difference[s] - cost\_difference\_cond2[s] + max\_diff\cdot cond2[s] <= max\_diff;$$

  zauważmy, że jeżeli $cond2[s] = 0$ to zachodzi trywialna nierówność

  $$cost\_difference[s] <= max\_diff,$$

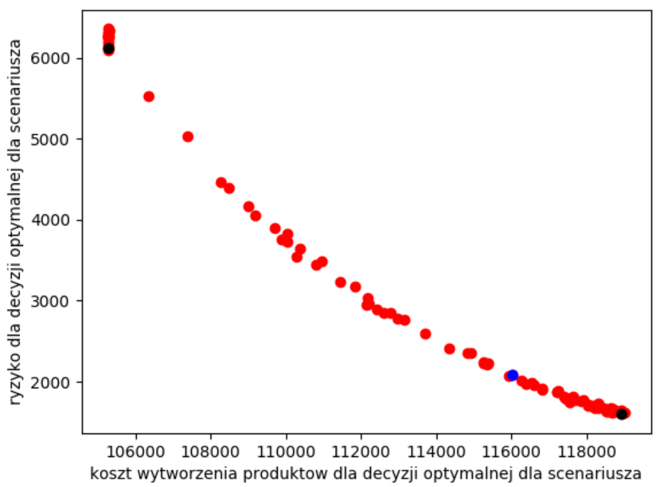
  a jeżeli $cond2[s] = 1$ wtedy $cost\_difference[s] = cost\_difference\_cond2[s]$ i otrzymujemy prawdę

  $$max\_diff <= max\_diff.$$

2.4 Wyznaczenie zbioru rozwiązań efektywnych

Zbiór rozwiązań efektywnych wyznaczono obliczając optymalne decyzje produkcyjne dla różnych parametrów awersji do ryzyka λ. Rozwiązania efektywne to rozwiązania niezdominowane przez żadne inne rozwiązania w modelu preferencji zadanym przez skalaryzację. Zdecydowano się rozważyć 100 wartości λ równo rozłożonych na odcinku od 0 do 10.

Otrzymany zbiór rozwiązań w przestrzeni ryzyko-koszt przedstawia Rysunek 1.



Rysunek 1: Zbiór rozwiązań w przestrzeni ryzyko-koszt. Koszt opisano na osi X, a ryzyko na osi Y. Kolorem czerwonym oznaczono znalezione rozwiązania. Kolorem czarnym oznaczono rozwiązania efektywne minimalnego kosztu i minimalnego ryzyka. Kolorem niebieskim oznaczono trzecie rozwiązania wykorzystane do liczenia dystrybuant.

2.5 Rozwiązania efektywne minimalnego ryzyka i kosztu

Rozwiązania efektywne minimalnego kosztu (minimalnego ryzyka), to niezdominowane rozwiązania dla których koszt (ryzyko) jest minimalne. Na Rysunku 1 są to punkty oznaczone kolorem czarnym. Rozwiązanie o minimalnym koszcie znajduje się w lewym górnym rogu rysunku. Rozwiązanie o minimalnym ryzyku znajduje się w prawym dolnym rogu. Konkretne wartości rozwiązań w przestrzeni ryzyko-koszt wynoszą:

- rozwiązanie efektywne minimalnego kosztu ma wartości ($\lambda = 0.3$):

$$(koszt, ryzyko) = (105263.37439251498, 6118.815806644055)$$

- rozwiązanie efektywne minimalnego ryzyka ma wartości ($\lambda = 9.8$):

$$(koszt, ryzyko) = (118928.72471366821, 1598.2189098201857)$$

2.6 Dominacja stochastyczna pierwszego rzędu

Do zbadania stochastycznej dominacji wybrano trzy rozwiązania efektywne: rozwiązanie minimalnego kosztu, rozwiązanie minimalnego ryzyka i trzecie, przypadkowo wybrane rozwiązanie. Wybrane rozwiązania mają następujące wartości w przestrzeni koszt-ryzyko:

1. R_1 - rozwiązanie efektywne minimalnego kosztu ma wartości ($\lambda = 0.3$):

$$(koszt, ryzyko) = (105263.37439251498, 6118.815806644055)$$

2. R_2 - rozwiązanie efektywne minimalnego ryzyka ma wartości ($\lambda = 9.8$):

$$(koszt, ryzyko) = (118928.72471366821, 1598.2189098201857)$$

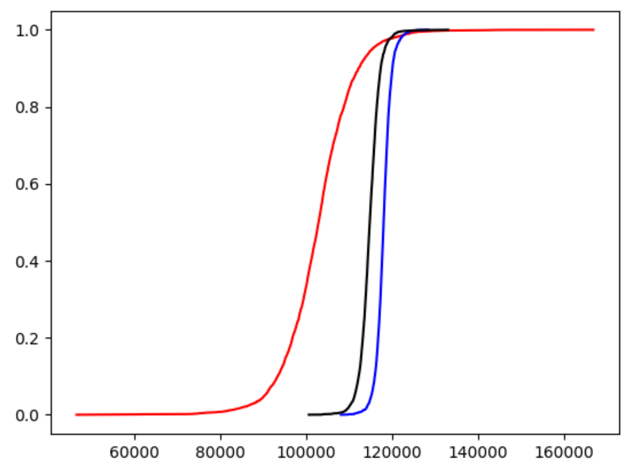
3. R_3 - trzecie rozwiązanie efektywne ($\lambda = 5$):

$$(koszt, ryzyko) = (115922.93830447218, 2077.73317791873718)$$

Zgodnie ze skryptem z wykładu dominację stochastyczną pierwszego rzędu opisuje Definicja 2.1.

Definicja 2.1 (Dominacja stochastyczna pierwszego rzędu)**.** *Zmienna losowa Y' dominuje zmienną losową Y'' w sensie dominacji stochastycznej pierwszego rzędu jeżeli dla każdego $v \in R$ prawdziwa jest nierówność dystrybuant $F_{Y'}(v) \leqslant F_{Y''}(v)$ i dla przynajmniej jednej wartości v jest to nierówność ostra.*

Napisano program obliczający dystrybuanty dla trzech wybranych rozwiązań efektywnych. Dystrybuanty obliczono po uporządkowaniu wartości kosztów od malejąco, ponieważ rozważamy problem minimalizacji. Następnie porównano wartości dystrybuant we wszystkich punktach odpowiadających możliwym kosztom. Nie otrzymano dominacji stochastycznej między żadną parą dystrybuant. Wykres dystrybuant zaprezentowano na Rysunkach 2 oraz 3.

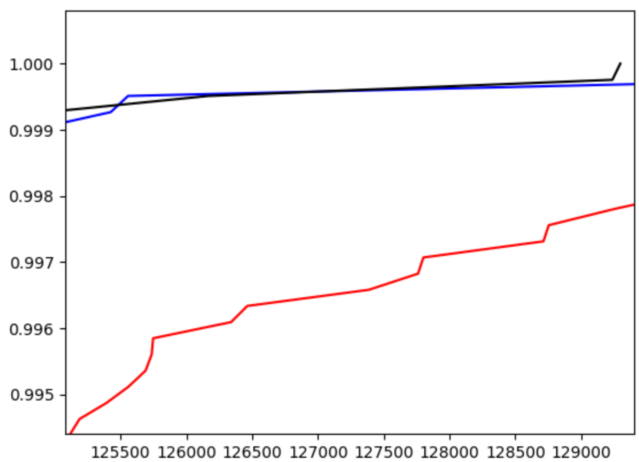


Rysunek 2: Dystrybuanty rozwiązań R_1 (kolor czerwony), R_2 (kolor niebieski) i R_3 (kolor czarny).

3 Testy poprawności implementacji

Stworzono jednostkowe testy poprawności implementacji:

- sprawdzanie czy dla wybranej decyzji wszystkie ograniczenia są spełnione,
- test obliczania wartości oczekiwanej (zgodnie z wektorem testowym podanym dla trójwymiarowego rozkładu t-Studenta w materiałach do projektu),
- test sprawdzający czy prawdopodobieństwa scenariuszy sumują się do jedynki.



Rysunek 3: Dystrybuanty rozwiązań R_1 (kolor czerwony), R_2 (kolor niebieski) i R_3 (kolor czarny).